

**DRAFT**  
**APPLICATION SECURITY**  
**ASSESSMENT TOOL**  
**MARKET SURVEY**



**Version 1.0**  
**October 1, 2002**

**Applications and Computing Security Division**  
**Center for Information Assurance Applications**  
**5275 Leesburg Pike**  
**Falls Church, VA 22041**

(This document is for review. Comments, if any, can be sent to [JainD@ncr.disa.mil](mailto:JainD@ncr.disa.mil)  
or [KoehlerS@ncr.disa.mil](mailto:KoehlerS@ncr.disa.mil))

**FOR INFORMATIONAL PURPOSES**

## TABLE OF CONTENTS

Assessment Tool .....	i
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Scope .....	1
1.3 Intended Audience.....	2
1.4 Document Structure.....	2
2. Methodology.....	3
2.1 Discussion .....	3
2.2 Capability Definitions .....	5
3. Tables.....	10
3.1 Web Application ASAT Table.....	10
3.2 Database ASAT Table.....	17
3.3 Developer ASAT Table.....	21
3.4 General-Purpose ASAT Table .....	26
4. Reviews.....	30
4.1 Web Application ASATs .....	30
4.1.1 AppScan.....	30
4.1.2 DominoScan.....	31
4.1.3 HailStorm.....	32
4.1.4 N-Stealth.....	33
4.1.5 Nikto.....	34
4.1.6 ScanDo .....	35
4.1.7 WebEnforcer .....	36
4.1.8 WebInspect.....	36
4.1.9 Web Scarab .....	37
4.1.10 WebSleuth.....	38
4.1.11 whisker (and libwhisker).....	39
4.1.12 WhiteHat Arsenal.....	40
4.2 Database ASATs .....	41
4.2.1 AppDetective .....	41
4.2.2 Database Scanner .....	42
4.2.3 OraScan.....	43
4.3 Developer ASATs .....	45
4.3.1 BFBTester .....	45
4.3.2 CLint .....	46
4.3.3 Cqual.....	46
4.3.4 FlawFinder .....	47
4.3.5 Fuzz.....	48
4.3.6 ITS4.....	48
4.3.7 Jlint.....	49
4.3.8 PyChecker .....	50
4.3.9 RATS .....	50
4.3.10 Splint .....	51
4.4 General-Purpose ASATs.....	53
4.4.1 Bv-Control .....	53

**Draft**

4.4.2 FoundScan..... 54  
4.4.3 Nessus ..... 55  
4.4.4 NetRecon..... 56  
4.4.5 Retina ..... 57  
4.4.6 SAINT..... 58  
4.4.7 Security Analyzer..... 59  
4.4.8 STAT Scanner..... 60  
4.4.9 Typhon II..... 61  
4.5 Other Useful Tools ..... 63  
4.5.1 Achilles ..... 63  
4.5.2 AppShield ..... 64  
4.5.3 Blast ..... 64  
4.5.4 IDA Pro Disassembler ..... 65  
4.5.5 Metis ..... 65  
4.5.6 PatchLink Update..... 65  
4.5.7 SecurityExpressions ..... 66  
4.5.8 STAT Analyzer..... 67  
4.5.9 STAT Neutralizer..... 67  
4.5.10 StormWatch and StormFront ..... 68  
4.5.11 WebProxy..... 68  
5. Conclusion ..... 70  
6. References ..... 71

## **1. INTRODUCTION**

### **1.1 PURPOSE**

Application security assessment tools (ASAT) are proactive tools that can be used to identify vulnerabilities in applications before they can be exploited by a malicious entity. ASATs will identify a scanned platform's major software applications and will match each application with known vulnerabilities associated with that application. Most ASATs employ large databases of known vulnerabilities to which they refer in their effort to identify vulnerabilities associated with commonly used operating systems and applications. Some ASATs specialize in identifying vulnerabilities within a specific type of application set, such as databases and their accompanying servers, hosts, and interfaces. A new breed of ASATs is emerging that is specifically geared toward assessing Web application security. These tools differ from traditional ASATs in that they do not perform many of the conventional assessments (such as port or host scanning); instead, they focus their search on vulnerabilities such as buffer overflows, potential field manipulation, and cookie poisoning. Additionally, increasing emphasis is being placed on the use of secure source coding practices in software development. As a result, new assessment tools have been developed to help identify potential vulnerabilities in source code in order to provide a reasonable starting point for performing manual security audits before deploying an application.

This document provides a broad, high-level survey of the many types of ASATs available on the market, including those designed for databases, Web applications, and source code. Tools identified in the survey are categorized and each tool's various attributes (e.g., usability, technical characteristics, and operational characteristics) are reviewed. Attributes are presented in a tabular format by category, and each tool is individually described and annotated.

This Application Security Assessment Tool Market Survey is the third document in a series being prepared by the Applications and Computing Security Division, Center for Information Assurance Applications. The first document establishes a set of application security requirements, which the second document addresses in the form of a developer's guide designed to assist developers with requirements implementation. The third and fourth documents in this series will identify application security assessment tools and present accompanying assessment methodologies for their use.

### **1.2 SCOPE**

This document is a "living document" that will be updated periodically and as new ASATs are developed and made available. All of the tools reviewed in this document were either available for use or slated for imminent release as of July 12, 2002. Although this survey is not by any means all-inclusive, it is intended to be as comprehensive as possible. Note that this document is not intended to be a "buyer's guide" or to recommend one particular tool over another, instead, it should be used for informational

## Draft

purposes, to further enlighten the reader as to the types and functionality of the tools available.

### 1.3 INTENDED AUDIENCE

Application developers should use this document as a starting point for testing their applications so they function securely on Department of Defense (DoD) systems. The document will help familiarize application developers with the types of tools available to them to assess vulnerabilities at various points in the application development life cycle. The tools presented in this guide will assist the developer in understanding what vulnerabilities are present in their applications and possibly what remedies are available to correct or mitigate those vulnerabilities.

### 1.4 DOCUMENT STRUCTURE

This document consists of six sections. An overview of these sections is provided below.

- **Section 1, Introduction**—describes the objectives, scope, and structure of this document.
- **Section 2, Methodology**—presents the approach used to developing this document.
- **Section 3, Tables**—provides a tabularized comparison view of the various tools reviewed in this document.
- **Section 4, Reviews**—presents a brief description and commentary for each tool reviewed.
- **Section 5, Conclusion.**
- **Section 6, References.**

## 2. METHODOLOGY

### 2.1 DISCUSSION

The following steps were used to conduct the ASAT market survey.

Step 1. A thorough review was made of available ASATs using magazines, journals, online publications, and other information available on the Internet. Information on all varieties of tools was gathered, with particular emphasis placed on currently available (and supported) products or those nearing completion. Literature was reviewed to provide an extensive list of available vulnerability assessment tools.

Step 2. The extensive list of ASATs that was obtained through the above review was then divided into four categories, organizing the discovered tools according to their intended functionality. A fifth category was added for tools that, although not directly used for vulnerability assessment, were deemed interesting and were thought to be able to play an important role in the overall application security process. This fifth category is not meant to be exhaustive. Definitions for the five categories are listed below.

- **Web Application ASATs.** These tools are designed to automatically scan Web applications, sites, and servers, seeking for potential vulnerabilities. These tools differ from general vulnerability assessment tools (see step 4) in that they do not perform a broad range of checks on a myriad of software and hardware (i.e., port scanning or host vulnerability scanning). Instead, they perform other checks, such as potential field manipulation and cookie poisoning, which allows a more focused assessment of Web applications by exposing vulnerabilities that standard Vulnerability and Assessment (VA) tools are unaware of.
- **Database ASATs.** These tools are hybrid scanners designed to specifically evaluate and assess database vulnerabilities. These tools perform both penetration testing and auditing, generally scanning for known configuration vulnerabilities, incorrect settings, weak security profiles, and missing patches/out-of-date software.
- **Developer ASATs.** These tools are designed to locate potential vulnerabilities in source code or compiled programs. They do not definitively find bugs; rather, they provide a reasonable starting point for performing manual security audits. These tools are composed of the following:
  - **Source-Code Scanners.** These mainly open-source tools are very new and generally still under development. Their purpose is to scan source code, to identify potentially dangerous function calls.
  - **Fuzzers and Buffer Overflow Generators.** These tools test software by bombarding the program with random data. Note that most of these tools are open source applications.

## Draft

- **General-Purpose ASATs.** These tools scan networks and systems for potential security weaknesses and recommend fixes. Because these tools are designed to scan a broad range of software and hardware, their ability to focus in depth on the vulnerabilities of a specific item, such as a database, is limited.
- **Other.** This is a catchall category to capture interesting and potentially useful tools that may aid in providing application and database security. This category is composed of a range of tools, which are not directly used for vulnerability assessments, but may either assist in securing a site or complement the vulnerability assessment process.

Step 3. After an initial categorization of discovered ASATs, a set of capability definitions was established to improve the ability to compare and review tools in a similar manner. Because of the broad nature of the tools, it was difficult to compare Web Application ASATs with Database ASATs. Therefore, it was decided that comparisons with similar types of tools be limited when possible. Section 2.2 outlines the capability definitions used in the market survey. Note that some capabilities apply to a specific category of tool and that not all capabilities were used when reviewing all tools.

Step 4. Each tool was then assessed based on a review of literature available. Note that the vendor or developer of the tool provided a majority of the information reviewed, in the form of product specifications or user-manuals/documentation. In addition, if a demonstration/evaluation version of the software was easily available, suitable copies were downloaded and installed to further evaluate the tool.

Step 5. Each tool was reviewed using the capability definitions developed in Section 2.2. The results of this review were tabulated into a matrix based on category-type of tool. These tables are presented in Section 3. Because of the nature of this market survey, every reasonable attempt was made to accurately assess via the capability definitions each tool based either on available literature or hands-on experimentation.

Step 6. Each tool was further reviewed and documented in Section 4. A description of each tool and its functionality is provided along with a small commentary.

*Note that this document is a market survey. Although it is extensive, it is not complete because tools are continually being developed or discontinued. Moreover, this document does not favor one over another, nor is it intended to be used as a “buyer’s guide.” It is for informational purposes only.*

## 2.2 CAPABILITY DEFINITIONS

The survey attempted to assess each tool along the following list of capabilities.<sup>1</sup> Table 2-1 lists the capability categories and definitions.

**Table 2-1 Capability Definitions**

NUMBER	CAPABILITY	DESCRIPTION
<b>GENERAL INFORMATION</b>		
1.1	Name/Version	The specific tool and its version number
1.2	Vendor or Source of Tool	The name of the vendor or source of the software (if open-source)
1.3	Cost	Costs. Licensing Requirements. Maintenance Fees. (See Capability 4.2.1 for additional restrictions, if any.)
1.4	Scanner Platform/Architecture	The type of architecture, and the hardware and software required to operate the vulnerability assessment tool.
1.5	Scanning Targets	The types of platforms and network elements subject to assessment by each tool will be identified.
1.6	Assurance	NIAP certified or DITSCAP or NIACAP reviewed
<b>USABILITY ATTRIBUTES</b>		
2.1	<b>Ease of Use</b>	
2.1.1	Installation, Updates, and Maintenance	Methods of installation, updates, and maintenance.
2.1.2	Configuration	Method of configuration for testing
2.1.3	Intuitive Graphical Used Interface (GUI)	This means that there are well defined pull down menus and labeled buttons that are all descriptive enough to make navigation of the GUI and the available functions easy
2.1.4	Crash Recovery	This relates to any backup procedures, auto save functions, or recovery operations that are possible if a system crashes. Basically, any function that prevents the loss of data in the event of a crash
2.1.5	Emergency STOP	In the event testing causes any adverse conditions, there should be an easy method to immediately stop all testing operations
2.1.6	Methods for running tasks	Automated, event-driven, scheduled, manual, or any other type of method for running a test
2.1.7	Ability to save profiles or sessions	When custom configurations containing testing details are created, the ability to save these configurations and then re-run them exists within the tool. Also, the ability to use other sessions as the basis of new custom configurations.

<sup>1</sup> Note that this capability list borrows from, as well as modifies, the National Security Agency's Vulnerability Assessment Tool Comprehensive Capabilities List (VATCCL) Release 0.2, dated February 5, 2002.

**Draft**

NUMBER	CAPABILITY	DESCRIPTION
2.1.8	Potential to cause damage limited or clearly defined	If a tool can cause damage, all testing functions that can cause damage are clearly identified. If needed, these testing functions are also limited.
<b>2.2 Support</b>		
2.2.1	Type of Support	This assistance can either be a combination of a Customer Help Desk during business hours and access to technical solutions through the vendor's Web site.
2.2.2	Updates	Product and signature update availability. Are these updates automatic? How frequent are these updates released? Does the vendor notify users when updates become available? Is there any cost involved with any of the above?
2.2.3	Bug Fixes and Client Input	Often the client will find bugs, errors, or confusion in the product and associated documentation. When this occurs, is there any easy method for reporting these problems to the vendor, and if so are these issues corrected or responded to in a timely manner?
<b>2.3 Documentation</b>		
2.3.1	Complete	Help files, support documentation. All areas of the tool and its functionality are included in the documentation
2.3.2	Updated in a Timely Manner	Documentation is updated as changes in the product occur
<b>TECHNICAL ATTRIBUTES</b>		
<b>3.1 Configuration Customization</b>		
3.1.1	Allows the use of scripting to customize the scanning application, its program modules, or vulnerability scan tests	Scanner allows the customer to use scripting to customize modules and integrate them into the vulnerability database based on their unique needs. Scanner allows the customer to use scripting to create custom scan tests for any Internet Protocol (IP) device or protocol.
3.1.2	More than one scripting language can be used	Scanner allows the user to use more than one scripting language, preferably not a vendor proprietary language. If it is, it should be easily used and not require additional training.
3.1.3	Can be tailored for multiple environment configurations of scan and vulnerability settings.	Scanner allows the user to set up and save specific profiles for multiple operating environments (i.e., profiles are tailored to check for those vulnerabilities associated with only a particular environment. Once the profile is set up, the user does not have to change what is scanned each time it is run.)
3.1.4	Supports fine-tuning of scanner (e.g., minimize false-positive/false-negative conditions).	Test report will indicate if this is a capability and any limitations.
3.1.5	Provides a capability to modify system scan profiles, including multiple criteria.	Administrators should have the option to manually determine what systems not to scan. (For example, the scanner can be configured to scan for both security

**FOR INFORMATIONAL PURPOSES**

**Draft**

NUMBER	CAPABILITY	DESCRIPTION
		policy and a specific Operating System [OS] in one scan session).
<b>3.2</b>	<b>Vulnerabilities Scanned</b>	
<b>3.2.1</b>	Check for known vulnerabilities associated with multiple OS's.	This includes checks for specific vulnerabilities associated with some of the more popular OS's (e.g., NT, UNIX,).
<b>3.2.2</b>	Check for known vulnerabilities associated with various applications.	Supports emerging trend in specialized scanners for databases (i.e., Oracle, Structured Query Language [SQL], Sybase) and popular applications.
<b>3.2.3</b>	Check for known vulnerabilities associated with Web server vulnerabilities.	Includes checks for specific vulnerabilities embedded in particular Web servers.
<b>3.2.4</b>	Check for known vulnerabilities associated with mobile code.	E.g., ActiveX, Java applets/applications.
<b>3.2.5</b>	Check for known vulnerabilities associated with browsers.	Examines vulnerabilities related to browsers.
<b>3.2.6</b>	Check for vulnerable software programs.	Checks for other known vulnerable programs. It verifies versions and flags needed patches and out-of-date files.
<b>3.2.7</b>	Checks for network device configurations vulnerabilities.	E.g., Printers, routers, and firewalls.
<b>3.2.8</b>	Checks for known vulnerabilities associated with network services and protocols.	The test report will enumerate which services and protocols the scanner identifies (e.g., http, ftp, telnet, SMTP).
<b>3.2.9</b>	Checks for the SANS Top Vulnerabilities.	Ties in with the SANS Top Vulnerabilities and can be configured to check for these either singularly or in conjunction with other module scans.
<b>3.2.10</b>	Capable of discovering vulnerabilities not previously identified.	Focuses on potential future capability.
<b>3.2.10.1</b>	Cookie Poisoning	Changing the cookie's contents
<b>3.2.10.2</b>	Hidden Field Manipulation	Changing hidden fields in a page's source code
<b>3.2.10.3</b>	Parameter Tampering	Changing information in a site's Uniform Resource Locator (URL) Common Gateway Interface (CGI) parameters
<b>3.2.10.4</b>	Buffer Overflow Attacks	Sending large request messages to the application
<b>3.2.10.5</b>	Cross-Site Scripting	Malicious code injected into a site that tricks legitimate users into believing it originated from the actual site
<b>3.2.10.6</b>	Backdoor and Debug Options	Exploiting vulnerabilities left open in internally developed code
<b>3.2.10.7</b>	Forceful Browsing	Subverting the application flow of a web page
<b>3.2.10.8</b>	Stealth Commanding	Planting Trojan horses in text fields that cause the Web application to perform unintended commands
<b>3.2.10.9</b>	3 <sup>rd</sup> Party Misconfiguration	Exploiting configuration errors in third party components
<b>3.2.10.10</b>	Database Sabotage (SQL Injection)	Concatenating various SQL commands to input fields to affect the regular operation of the database

**FOR INFORMATIONAL PURPOSES**

**Draft**

NUMBER	CAPABILITY	DESCRIPTION
3.2.10.11	Data Encoding	Sending requests using different data encoding standards
3.2.10.12	Protocol Piggyback	Modifying the application protocol structure
3.2.11	Capable of discovering potential vulnerabilities in Source Code	These tools are designed to located potential vulnerabilities in either source-code or compiled programs. They do not definitively find bugs; rather, they provide a reasonable starting point for performing manual security audits. (Note that this capability applies solely to developer ASATs).
<b>3.3</b>	<b>Requirements Scanned (Policy and Procedure Compliance)</b>	
3.3.1	Checks passwords for strength, complexity, and compliance with established security policy.	Includes checks for secure passwords. These checks could include testing for passwords that might be guessed using common schemes, dictionary passwords, missing passwords, default passwords, disabled passwords, blank passwords, guest accounts, and no expiration date for passwords.
3.3.2	Checks for unusual account activity.	Includes analysis of user activities; checks for unused accounts; checks for sniffer activity and NT Network monitoring; checks for unusual filenames and misplaced files; and checks for GetAdmin privileges.
3.3.3	Identifies the groups and accounts associated with those groups.	Includes checking for proper group membership and privileges, checks for invalid or duplicate Group Identification (GIDs), checks for root-equivalents and nonexistent accounts and other group-related issues.
3.3.4	Checks account privileges.	Examines user privileges. These privileges might include checks for invalid User Identification (UIDs) and GIDs, checks for root-equivalent accounts, and checks for unauthorized users possessing the ability to debug programs.
3.3.5	Identifies inactive accounts.	Identifies accounts that have been inactive for N number of days.
3.3.6	Check of system auditing.	Encompasses checks for properly configured system auditing.
3.3.7	Check for key lengths in browsers and other applications.	Browser and application key lengths are identified.
<b>OPERATIONAL ATTRIBUTES</b>		
<b>4.1</b>	<b>REPORTING</b>	
4.1.1	Parse/Normalize/Query/Analytical Functions	These are features in the reporting that allow searching, sorting, and other type of querying that will allow for analysis. These are also features that would allow for an automated analysis.
4.1.2	Prioritize Data and Data Reduction	These are features that allow for the prioritization in reporting of the findings based on a selection of the criteria for prioritization. These are also features that allow for data sets to be reduced in size based on parameters and their values.
4.1.3	Data Visualization	These are features that allow for a graphical interpretation of the data that has been collected.

**FOR INFORMATIONAL PURPOSES**

**Draft**

<b>NUMBER</b>	<b>CAPABILITY</b>	<b>DESCRIPTION</b>
4.1.4	Import and Export Formats that are supported	A listing of the formats that may be imported into the tool for reporting purposes and the formats that the reports and data can be exported as.
4.1.5	Differential Reporting and Trend Analysis	Differential reporting allows for reports from different times on similar systems to be compared so that differences are identified. Trend analysis allows for multiple reports over time to be analyzed to determine any trends.
4.1.6	Report generation templates, creation and customization of templates, and fine tuning of report outputs	Does the tool allow for easy creation of templates to generate specific and custom reports? In addition, does this or any other methods allow for the fine-tuning of what will be included in the final reports that are generated?
4.1.7	Auto Fix or Fix Information	These features include either a way to automatically repair and patch vulnerabilities that are discovered, or complete documentation on the vulnerability (or potential vulnerability) that has been discovered so that the appropriate patch can be applied or designed so the vulnerability is fixed.
<b>4.2</b>	<b>SECURITY</b>	
4.2.1	Access to the tool	A control method is in place that only allows authorized users access to the tool.
4.2.2	Logging of Users and their actions	A logging method is in place that is inaccessible by users that will log not only who has used the tool, but also what actions or commands were run by the user
4.2.3	Tampering and Integrity Control	Methods are in place to ensure the tool, signatures, documentation, or any other aspect of the tool has been tampered with or replaced with Trojans or erroneous data of any type. This can include such methods as checksums and signed updates.
4.2.4	Tools or Users access to targets	Limiting factors are in place that either restrict the tool or the user to scanning only acceptable targets or target ranges.

### **3. TABLES**

The tables in this section were developed using Microsoft Excel. A table for each of the first four categories (Web Application, Database, Developer, and General Purpose) is a separate worksheet in the same file. These tables are provided as a separate electronic attachment. [Note: For convenience, the Application Security Assessment Tool worksheets have been pasted in the sections below. To view the entire worksheet with survey notes that correspond to specific cells, you will need to refer to the Microsoft Excel file (Market Survey Matrix 72502.xls)] Also note that comments have been inserted into many of the cells within each spreadsheet. These comments might not appear in printed version of each table, but will always appear in the electronic version. (If a cell contains a comment, it is denoted with a small red tab in the upper-right corner of the cell.)

#### **3.1 WEB APPLICATION ASAT TABLE**

Table 3-1 presents Web application ASATs identified while conducting the market survey. The table contains a mapping of each tool’s capabilities to the capability definitions presented in Section 2.2.

*[Note: For convenience, the Web Application Tool worksheet has been pasted in the pages below. To view the entire worksheet with survey notes that correspond to specific cells, you will need to refer to the Microsoft Excel file (Market Survey Matrix 72502.xls) and the worksheet labeled “Web Application.”]*

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

Number	Criteria	AppScan	DominoScan	Hailstorm	N-Stealth	Nikto	ScanDo	WebEnforcer
<b>1 General Information</b>								
1.1	<b>Name / Version</b>	AppScan (v3.0, released in June 2002)	Next Generation Software DominoScan 1.1	Cenzic Hailstorm 3.0	N-Stealth	Nikto (v1.10)	Kavado ScanDo	HP WebEnforcer 2.0
1.2	<b>Vendor or Source of Tool</b>	Sanctum (www.sanctuminc.com)	Next Generation Software	Cenzic Inc., formerly known as ClickToSecure	N-Stalker	Chris Sullo (http://www.cirt.net/code/nikto.shtml)	Kavado	Hewlett Packard
1.3	<b>Cost</b>	\$15000 plus an additional annual maintenance fee (License restricts use to certain IPs)	Unknown	\$30,000	Variable based on IPs licensed, See Comment	GNU General Public License -- Open Source	Unknown	\$2,995
1.4	<b>Scanner Platform/Architecture</b>	Windows 2000	Windows	Windows 2000 or Windows XP with IE6 and MSSQL 2000, Intel Pentium III, 500MHz, 512MB RAM	Windows 95/98/ME/NT/2000/XP	Windows, Unix, Linux	Windows NT, Windows 2000	Windows NT, Windows 2000
1.5	<b>Scanning Targets</b>	Any Web Application	Domino Web Server	All types of network applications and hardware	Web Servers	Web Servers and Sites	All types of Web servers	Components of Windows Web Server Environment (Only on same machine installed on)
1.6	<b>Assurance</b>		No Known Certifications	No Known Certifications	No Known Certification	None	No Known Certifications	No Known Certifications
<b>2 Usability Attributes</b>								
<b>2.1 Ease of Use</b>								
2.1.1	Installation, Updates, and Maintenance	Wizard-Driven Installation	Simple	GUI Based with detailed instructions at each step	Simple	Simple. Requires Perl.	Unknown	Easy Installation, subscription based updates
2.1.2	Configuration	GUI	GUI	Simple Configuration, GUI	Simple Configuration, GUI	Simple. Via modification of config files	GUI	Simple Configuration, GUI
2.1.3	Intuitive GUI	Yes, via Wizards	Yes	For the most part, %95 Yes	Partially	Command-line driven	Yes	Yes
2.1.4	Crash Recovery		Unknown, doesn't appear to	Unknown, doesn't appear to	No	No	Unknown, doesn't appear to	No
2.1.5	Emergency STOP	Yes	Yes	Yes	Yes	No	Yes	Unknown, trial ordered
2.1.6	Methods for running tasks	Automated and Manual	Manual	Manual, Scheduled	Manual	Manual	Manual, Multiple Scans at same time	Manual, Automatic Monitoring
2.1.7	Ability to save profiles or sessions		No	Yes	No	Yes	Yes	Wizards to create and save profiles
2.1.8	Potential to cause damage limited or clearly defined	Yes	Yes	Very advanced tool with testing procedures that are potentially damaging	No	Yes	Yes	Yes
<b>2.2 Support</b>								
2.2.1	Type of Support	Customer Service and Training	Unknown	Email based support, 9AM to 9PM EST, phone # available	Yes	via web-site	email and online FAQ	Not included, must purchase separate Support Agreement
2.2.2	Updates	Via AppScan Extranet	Unknown	Access to database of new patterns and test, when available	By Subscription, Automatic	automated via web-site	unknown	With purchase of separate Support Agreement, subscription based
2.2.3	Bug Fixes and Client Input	Bug fixes included in annual license fee	Unknown	Bug reports and issues assigned within 8 hours of a request	Yes	via web-site	Claim timely, but time unknown	Response time unknown
<b>2.3 Documentation</b>								
2.3.1	Complete	Yes	Limited online, waiting for trial to arrive	Yes	Limited	Limited	Limited online, waiting for trial to arrive	Limited online, waiting for trial to arrive
2.3.2	Updated in a Timely Manner	Yes	Unknown	Yes	Unknown	Unknown	Unknown	Unknown
<b>3 Technical Attributes</b>								
<b>3.1 Configuration Customization</b>								
3.1.1	Allows the use of scripting to customize the application, its program modules or scan tests	No	No	Yes	Yes	Yes	Yes	No
3.1.2	More than one scripting language can be used	N/A	No	Yes	No	No	Yes	No

**FOR INFORMATIONAL PURPOSES**

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

3.1.3	Can be tailored for multiple environment configurations of scan and vulnerability settings.	Yes	Limited	Yes	Yes	Yes	Yes	Yes
3.1.4	Supports fine-tuning of scanner (e.g., minimize false-positive/false-negative conditions).	Yes	No	Yes	Yes	Yes	Yes	Yes
3.1.5	Provides a capability to modify system scan profiles, including multiple criteria.	Yes	Limited	Yes	Yes	Yes	Yes	Yes
<b>3.2 Vulnerabilities Scanned</b>								
3.2.1	Check for known vulnerabilities associated with multiple operating systems.	No	No	Yes	No	Yes	No	No
3.2.2	Check for known vulnerabilities associated with various applications.	Yes	No	Yes	No	Yes	No	No
3.2.3	Check for known vulnerabilities associated with web server vulnerabilities.	Yes	Yes, Domino Webserver only	Yes	Yes	Yes	Yes	Yes
3.2.4	Check for known vulnerabilities associated with mobile code.	Yes	No	Yes	No	Unknown	No	No
3.2.5	Check for known vulnerabilities associated with Browsers.	No	No	Yes	No	No	No	No
3.2.6	Check for vulnerable software programs.	Yes	No	Yes	No	Yes	No	No
3.2.7	Checks for network device configurations vulnerabilities.	No	No	Yes	No	Unknown	No	No
3.2.8	Checks for known vulnerabilities associated with network services/protocols.	Yes (Application Layer)	No	Yes	No	Unknown	No	No
3.2.9	Checks for the SANS Top Vulnerabilities.		No	No	Yes	Unknown	No	No
3.2.10	Capable of discovering vulnerabilities not previously identified.				Yes	Partial		
3.2.10.1	Cookie Poisoning		No		Unknown	Unknown	Yes	No
3.2.10.2	Hidden Field Manipulation	Yes	No		Unknown	Unknown	Yes	No
3.2.10.3	Parameter Tampering	Yes	No		Unknown	Unknown	Yes	No
3.2.10.4	Buffer Overflow Attacks	Yes	No	Yes	Yes	Unknown	No	No
3.2.10.5	Cross-Site Scripting	Yes	No	Yes	Unknown	Unknown	No	No
3.2.10.6	Backdoor and Debug Options	Yes	No		Unknown	Unknown	Yes	No
3.2.10.7	Forceful Browsing	Yes	No		Unknown	Unknown	No	No
3.2.10.8	Stealth Commanding	Yes	No		Unknown	Unknown	Yes	No
3.2.10.9	3rd Party Misconfiguration	Yes	No		Yes	Unknown	Yes	No
3.2.10.10	Database Sabotage (SQL Injection)	Yes	No	Yes	Unknown	Unknown	Yes	No
3.2.10.11	Data Encoding		No		Unknown	Unknown	Yes	No
3.2.10.12	Protocol Piggyback		No		Unknown	Unknown	Yes	No
3.2.11	Capable of discovering potential vulnerabilities in Source Code	No	No	No	No	No	No	No
<b>3.3 Requirements Scanned (Policy and Procedure Compliance)</b>								

**FOR INFORMATIONAL PURPOSES**

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

3.3.1	Checks passwords for strength, complexity, and compliance with established security policy.	Yes (via Customized Scan)	No	No	No	No	No	Unknown
3.3.2	Checks for unusual account activity.	No	No	No	No	No	No	Unknown
3.3.3	Identifies the groups and accounts associated with those groups.	No	No	No	No	No	No	Unknown
3.3.4	Checks account privileges.	No	No	No	No	No	No	Unknown
3.3.5	Identifies inactive accounts.	No	No	No	No	No	No	Unknown
3.3.6	Check of system auditing.	No	No	No	No	No	No	Unknown
3.3.7	Check for key lengths in browsers and other applications.	No	No	No	No	No	No	Unknown
<b>4 Operational Attributes</b>								
<b>4.1 Reporting</b>								
4.1.1	Parse/Normalize/Query/Analytical Functions	Yes	No	Yes	No	No	Unknown	Unknown
4.1.2	Prioritize Data and Data Reduction	Yes	No	Yes	Yes	No	Unknown	Unknown
4.1.3	Data Visualization	Yes	No	Yes	No	No	Yes	Unknown
4.1.4	Import and Export Formats that are supported	Export (CSV, Word, RTF, HTML, TIFF, Excel, PDF)	See Comment	See Comment	Yes	Export report as a text file	See Comment	Unknown
4.1.5	Differential Reporting and Trend Analysis	No	No	No	No	No	Yes	Yes
4.1.6	Report generation templates, creation and customization of templates, and fine tuning of report outputs	Yes	No	Yes	No	No	Yes	No
4.1.7	Auto Fix or Fix Information	Yes	Fix Information	Fix Information	Fix Information	No	Fix Information	Auto Fix and Fix Information
<b>4.2 Security</b>								
4.2.1	Access to the tool	No	No	Yes	No	No	No	No
4.2.2	Logging of Users and their actions	No	No	None	No	No	None	None
4.2.3	Tampering and Integrity Control	No	None	None	No	No	None	Yes
4.2.4	Tools or Users access to targets	License limits scans to a specific range of IP addresses	No	Yes	License	No	Yes	No

**FOR INFORMATIONAL PURPOSES**

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

WebInspect	Web Scarab	WebSleuth	whisker & LibWhisker	WhiteHat Arsenal
WebInspect 2.0 (v2.5 released on June 17, 2002)	Web Scarab	WebSleuth (v1.3)	Whisker (v1.4) & LibWhisker (v1.4)	WhiteHat Arsenal (1.05 Release)
SPI Dynamics	The Open Web Application Security Project	<a href="http://www.geocities.com/dzzie/sleuth/">http://www.geocities.com/dzzie/sleuth/</a>	Rain Forrest Puppy ( <a href="http://www.wiretrip.net">www.wiretrip.net</a> )	Community WhiteHat Security ( <a href="http://community.whitehatsec.com/wharsenal/">http://community.whitehatsec.com/wharsenal/</a> )
\$4995 per tested server plus a yearly maintenance fee	Open Source (still under development)	Open Source	Open Source	Open Source (Free for Registered Users)
Windows 98/NT/2000/XP	Platform Independent (100% Java based)	Windows	Windows, Unix, Linux	RedHat Linux 7.2 and Apache 1.3.22
Any Web Application	Web Applications	Any Web Site	Web Servers and Sites	Any Web Site
	None	No	No	No
Wizard-Driven Installation	unknown (project still in development)	Wizard-Driven Installer	Simple. LibWhisker Requires Perl	Simple Linux-based
GUI	unknown (project still in development)	GUI	Simple. Via Config files	GUI
Yes	unknown (project still in development)	No	Command-line driven	No
	unknown (project still in development)	N/A	No	N/A
	unknown (project still in development)	Yes	No	
	unknown (project still in development)		Manual and Automated	Automated and Manually Driven
	unknown (project still in development)		Yes	Yes (as XML log file)
	unknown (project still in development)	No	No	No
Customer Service and Training	unknown (project still in development)	Open Source via website	Open Source via website	Open Source via website
Daily via website	unknown (project still in development)	Open Source via website	Open Source via website	Open Source via website
	unknown (project still in development)	Open Source via website	Open Source via website	Input form provided as part of the tool
	unknown (project still in development)	Limited	Limited	Limited
	unknown (project still in development)	Version Change document included with each new release	Unknown	Documentation is provided with each new version release
Yes (using VBScript)	Planned	Yes, but very rudimentary	Yes. Via Perl	
No	Planned (probably Java and XML)	No	No	

**FOR INFORMATIONAL PURPOSES**

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

Yes	Planned	N/A	Yes	N/A
Yes	unknown (project still in development)	N/A	Yes	N/A
Yes	Planned		Yes	
Yes	unknown (project still in development)		Yes	
Yes	Planned	Assists with	Yes	Assists with
Yes	Planned	Assists with	Yes	Assists with
Yes	unknown (project still in development)	No	Unknown	No
	Planned	Assists with	Unknown	Assists with
Yes	Planned	No	Yes	No
	unknown (project still in development)	No	Unknown	No
	Planned	No	Unknown	No
	unknown (project still in development)	Assists with	Unknown	Assists with
			Partial (Can be modified)	
Yes	Planned	Assists with	Unknown	Assists with
Yes	Planned	Assists with	Unknown	Assists with
Yes	Planned	Assists with	Unknown	Assists with
Yes	Planned		Unknown	Assists with
Yes	Planned	Assists with	Unknown	Assists with
Yes	Planned		Unknown	Assists with
Yes	Planned	Assists with	Unknown	Assists with
Yes	Planned		Unknown	Assists with
Yes	Planned	Assists with	Unknown	Assists with
	Planned		Unknown	Assists with
	Planned		Unknown	Assists with
No	No	No	No	No

**FOR INFORMATIONAL PURPOSES**

**Draft**  
**Table 3-1 Web Application (Application Security Assessment Tools)**

No	No	Partial (w/ plugin)	No	No
No	No	No	No	No
No	No	No	No	No
No	No	Partial (w/ plugin)	No	No
No	No	No	No	No
No	No	No	No	No
No	unknown (project still in development)	No	No	No
Yes	unknown (project still in development)		No	
Yes	Planned	No	No	No
Yes	unknown (project still in development)	No	No	No
	unknown (project still in development)	Export report as a text file	Export report as a text file	
Yes	unknown (project still in development)	No	No	No
Yes	Planned		No	
Yes	Planned	No	No	No
No	unknown (project still in development)	No	No	No
No	unknown (project still in development)	No	No	No
No	unknown (project still in development)	No	No	No
One License per server to be tested	unknown (project still in development)	No	No	No

**FOR INFORMATIONAL PURPOSES**

### **3.2 DATABASE ASAT TABLE**

Table 3-2 presents database ASATs identified while conducting the market survey. The table contains a mapping of each tool's capabilities to the capability definitions presented in Section 2.2.

*[Note: For convenience, the Database Tool worksheet has been pasted in the pages below. To view the entire worksheet with survey notes that correspond to specific cells, you will need to refer to the Microsoft Excel file (Market Survey Matrix 72502.xls) and the worksheet labeled "Database."]*

**Draft**  
**Table 3-2 Database (Application Security Assessment Tools)**

Number	Criteria	AppDetective for Oracle Client	Database Scanner	OraScan
<b>1 General Information</b>				
1.1	<b>Name / Version</b>	AppDetective for Oracle Client	Database Scanner (v 4.2)	OraScan
1.2	<b>Vendor or Source of Tool</b>	Application Security Inc.	Internet Security Systems (www.iss.net)	Next Generation Software
1.3	<b>Cost</b>	\$1,295 per Oracle SID, SQL Server instance, Sybase Server, or Lotus Server		Not Yet Available (Scheduled release was to have been in May 2002)
1.4	<b>Scanner Platform/Architecture</b>	Windows NT 4.0 (SP5), 2000, XP	Windows 2000 (SP1) or Windows NT (SP6a)	unknown
1.5	<b>Scanning Targets</b>	Oracle 7, 8, 8i, 9i (Unix, MS Windows NT & 2000, Linux, Solaris). See Comment.	MS SQL Server 6.5, 7.0, 8.0 Oracle 7.3, 8.0.6, 8.1.7 Sybase 11.5, 11.9.2, 12.0	Oracle Databases
1.6	<b>Assurance</b>			unknown
<b>2 Usability Attributes</b>				
<b>2.1 Ease of Use</b>				
2.1.1	Installation, Updates, and Maintenance	Wizard-based Installation and Updates/Maintenance		unknown
2.1.2	Configuration	GUI	GUI	unknown
2.1.3	Intuitive GUI	Yes	Yes	unknown
2.1.4	Crash Recovery			unknown
2.1.5	Emergency STOP		Yes	unknown
2.1.6	Methods for running tasks	Automated, Scheduled and Manual	Automated	unknown
2.1.7	Ability to save profiles or sessions	Yes	Yes	unknown
2.1.8	Potential to cause damage limited or clearly defined	Yes	Yes	unknown
<b>2.2 Support</b>				
2.2.1	Type of Support	Tech Support and Customer Service via phone, fax, or e-mail	Tech support and emergency response support provided 24x7x365 via phone, and e-mail	unknown
2.2.2	Updates	Via subscription service	Automatic Notification of updates. Downloaded via a secure server.	unknown
2.2.3	Bug Fixes and Client Input	Via subscription service	Fixes part of the X-Press updates. Input encouraged via e-mail	unknown
<b>2.3 Documentation</b>				
2.3.1	Complete	Yes	Yes	unknown
2.3.2	Updated in a Timely Manner		With each new version and release.	unknown
<b>3 Technical Attributes</b>				
<b>3.1 Configuration Customization</b>				
3.1.1	Allows the use of scripting to customize the application, its program modules or scan tests	Yes. See Comment.	Yes	unknown
3.1.2	More than one scripting language can be used	No	No	unknown
3.1.3	Can be tailored for multiple environment configurations of scan and vulnerability settings.	Yes	Yes	unknown
3.1.4	Supports fine-tuning of scanner (e.g., minimize false-positive/false-negative conditions).	Yes	Yes	unknown
3.1.5	Provides a capability to modify system scan profiles, including multiple criteria.	Yes	Yes	unknown
<b>3.2 Vulnerabilities Scanned</b>				
3.2.1	Check for known vulnerabilities associated with multiple operating systems.	No	Yes, when used in conjunction with ISS Internet Scanner	unknown

**Draft**  
**Table 3-2 Database (Application Security Assessment Tools)**

Number	Criteria	AppDetective for Oracle Client	Database Scanner	OraScan
3.2.2	Check for known vulnerabilities associated with various applications.	Yes	Yes	unknown
3.2.3	Check for known vulnerabilities associated with web server vulnerabilities.	No	No	unknown
3.2.4	Check for known vulnerabilities associated with mobile code.	No	No	unknown
3.2.5	Check for known vulnerabilities associated with Browsers.	No	No	unknown
3.2.6	Check for vulnerable software programs.	Yes	Yes	unknown
3.2.7	Checks for network device configurations vulnerabilities.	No	No	unknown
3.2.8	Checks for known vulnerabilities associated with network services/protocols.	No	No	unknown
3.2.9	Checks for the SANS Top Vulnerabilities.	No	No	unknown
3.2.10	Capable of discovering vulnerabilities not previously identified.			
3.2.10.1	Cookie Poisoning	No	No	unknown
3.2.10.2	Hidden Field Manipulation	No	No	unknown
3.2.10.3	Parameter Tampering	No	No	unknown
3.2.10.4	Buffer Overflow Attacks	Yes	Yes	unknown
3.2.10.5	Cross-Site Scripting	No	No	unknown
3.2.10.6	Backdoor and Debug Options	No	No	unknown
3.2.10.7	Forceful Browsing	No	No	unknown
3.2.10.8	Stealth Commanding	No	No	unknown
3.2.10.9	3rd Party Misconfiguration	No	No	unknown
3.2.10.10	Database Sabotage (SQL Injection)	No	Yes	unknown
3.2.10.11	Data Encoding	No	No	unknown
3.2.10.12	Protocol Piggyback	No	No	unknown
3.2.11	Capable of discovering potential vulnerabilities in Source Code	No	No	unknown
<b>3.3 Requirements Scanned (Policy and Procedure Compliance)</b>				
3.3.1	Checks passwords for strength, complexity, and compliance with established security policy.	Yes. See Comment.	Yes. See Comment for examples.	unknown
3.3.2	Checks for unusual account activity.	No	No	unknown
3.3.3	Identifies the groups and accounts associated with those groups.	Yes	Yes	unknown
3.3.4	Checks account privileges.	Yes	Yes	unknown
3.3.5	Identifies inactive accounts.	No	Yes	unknown
3.3.6	Check of system auditing.	Yes	Yes	unknown
3.3.7	Check for key lengths in browsers and other applications.	No	No	unknown

**FOR INFORMATION PURPOSE**

**Draft**  
**Table 3-2 Database (Application Security Assessment Tools)**

Number	Criteria	AppDetective for Oracle Client	Database Scanner	OraScan
<b>4 Operational Attributes</b>				
<b>4.1 Reporting</b>				
4.1.1	Parse/Normalize/Query/Analytical Functions	Yes	Yes	unknown
4.1.2	Prioritize Data and Data Reduction	Yes	Yes	unknown
4.1.3	Data Visualization	Yes	Yes	unknown
4.1.4	Import and Export Formats that are supported	Export data to MS Access file. Reports can be saved in various formats such as .PDF. Scans and data can be imported.	Export to MS Word, MS Excel, Crystal Reports, Text and HTML	unknown
4.1.5	Differential Reporting and Trend Analysis	No	No	unknown
4.1.6	Report generation templates, creation and customization of templates, and fine tuning of report outputs	Wizard-based report creation, but no customization	No customization, but several types of standard reports are offered	unknown
4.1.7	Auto Fix or Fix Information	Yes, Fix information is provided	Yes, Fix information is provided	unknown
<b>4.2 Security</b>				
4.2.1	Access to the tool	No	No	unknown
4.2.2	Logging of Users and their actions	No	No	unknown
4.2.3	Tampering and Integrity Control	No	No	unknown
4.2.4	Tools or Users access to targets	No	No	unknown

### **3.3 DEVELOPER ASAT TABLE**

Table 3-3 presents developer ASATs identified while conducting the market survey. The table contains a mapping of each tool’s capabilities to the capability definitions presented in Section 2.2.

*[Note: For convenience, the Developer Tool worksheet has been pasted in the pages below. To view the entire worksheet with survey notes that correspond to specific cells, you will need to refer to the Microsoft Excel file (Market Survey Matrix 72502.xls) and the worksheet labeled “Developer.”]*

**Draft**  
**Table 3-3 Developer (Application Security Assessment Tools)**

Number	Criteria	BFBTester	CLint	Cqual	FlawFinder	Fuzz	ITS4	Jlint
<b>1 General Information</b>								
1.1	<b>Name / Version</b>	BFBTester	CLint	Cqual	FlawFinder	Fuzz	ITS4 (It's the Software, Stupid! (Security Scanner))	Jlint (v 2.3)
1.2	<b>Vendor or Source of Tool</b>	<a href="http://sourceforge.net/projects/bfbtester/">http://sourceforge.net/projects/bfbtester/</a>	<a href="http://sourceforge.net/projects/clint/">http://sourceforge.net/projects/clint/</a>	<a href="http://www.cs.berkeley.edu/~jfoote/cqual/">http://www.cs.berkeley.edu/~jfoote/cqual/</a>	David Wheeler ( <a href="http://www.dwheeler.com/flawfinder/">www.dwheeler.com/flawfinder/</a> )	<a href="http://sourceforge.net/projects/fuzz/">http://sourceforge.net/projects/fuzz/</a>	Cigital ( <a href="http://www.cigital.com/its4/">www.cigital.com/its4/</a> )	<a href="http://atho.com/jlint/">http://atho.com/jlint/</a>
1.3	<b>Cost</b>	GNU General Public License -- Open Source	GNU General Public License -- Open Source	GNU General Public License -- Open Source	GNU General Public License -- Open Source and Free Software	GNU General Public License -- Open Source	Open Source -- Non-Commercial License	GNU General Public License -- Open Source
1.4	<b>Scanner Platform/Architecture</b>	Solaris, FreeBSD, OpenBSD	UNIX, Linux	Linux	UNIX and Linux (author claims that it is portable to Windows). Requires Python.	UNIX, Linux	UNIX, LINUX, Windows	UNIX, Linux (Compiled with gcc) Windows (prepackaged with executable)
1.5	<b>Scanning Targets</b>	Binary programs	C++ Source Code	C Source Code	C and C++ Source Code	Binary programs	C and C++ Source Code	Java Source Code
1.6	<b>Assurance</b>	No	No	No	No	No	MD5 Checksum with each release	No
<b>2 Usability Attributes</b>								
<b>2.1 Ease of Use</b>								
2.1.1	Installation, Updates, and Maintenance	Makefile included	Makefile included	Makefile included	Makefile included. Runs as Python script	Makefile included		Makefile included
2.1.2	Configuration	Command-line driven	Command-line driven	Runs as part of the EMACS editor	Command-line driven -- runs as a Python script	Command-line driven	Command-line driven	Command-line driven
2.1.3	Intuitive GUI	Command-line driven	Command-line driven	Command-line driven. Runs as part of the EMACS editor	No	Command-line driven	Command-line driven	No
2.1.4	Crash Recovery	No	No	No	No	No	No	No
2.1.5	Emergency STOP	No	No	No	No	No	No	No
2.1.6	Methods for running tasks	Automated	Automated	Via initial annotation of source code -- then automated.	Automated	Automated	Automated and Manual	
2.1.7	Ability to save profiles or sessions		Yes	No	Yes -- as hitlists			Yes
2.1.8	Potential to cause damage limited or clearly defined	Implied by use	n/a	n/a	n/a	Implied by use	n/a	n/a
<b>2.2 Support</b>								
2.2.1	Type of Support	None	None	None	None	None	None	None
2.2.2	Updates	via web-site	via web-site	via web-site	via web-site	via web-site	via web-site	via web-site
2.2.3	Bug Fixes and Client Input	via web-site	via web-site	via web-site	via web-site	via web-site	via web-site	via web-site
<b>2.3 Documentation</b>								
2.3.1	Complete	Minimal	Complete	Complete	Minimal	Minimal	Minimal	Minimal
2.3.2	Updated in a Timely Manner	n/a	No	n/a	n/a	n/a	n/a	n/a
<b>3 Technical Attributes</b>								
<b>3.1 Configuration Customization</b>								
3.1.1	Allows the use of scripting to customize the application, its program modules or scan tests	No	No	Yes -- source code can be modified to accept arbitrary qualifiers	Program is a Python script and source is included, therefore it could be modified.	No	Vulnerability database is customizable	No
3.1.2	More than one scripting language can be used	No	No	No	No	No	No	No

**FOR INFORMATION PURPOSE**

**Draft**  
**Table 3-3 Developer (Application Security Assessment Tools)**

3.1.3	Can be tailored for multiple environment configurations of scan and vulnerability settings.	Command-line driven	Command-line driven	Command-line driven	Command-line driven	Command-line driven	Command-line driven	Command-line driven
3.1.4	Supports fine-tuning of scanner (e.g., minimize false-positive/false-negative conditions).	Partial	No	Yes, via code annotation	Yes	Partial	Yes	Partial -- Reporting of certain results may be disabled
3.1.5	Provides a capability to modify system scan profiles, including multiple criteria.	Partial	No	No	No	Partial	Yes	No
<b>3.2 Vulnerabilities Scanned</b>								
3.2.10	Capable of discovering vulnerabilities not previously identified.	Yes	Yes	Yes	Yes	Yes	Yes	Yes
3.2.11	Capable of discovering potential vulnerabilities in Source Code	Yes	Yes -- C++ Code	Yes -- C code	Yes -- C and C++ code	Yes	Yes -- C and C++ code	Yes - Java
<b>3.3 Requirements Scanned (Policy and Procedure Compliance)</b>								
		No	No	No	No	No	No	No
<b>4 Operational Attributes</b>								
<b>4.1 Reporting</b>								
4.1.1	Parse/Normalize/Query/Analytical Functions	No	No	Partial	Partial -- Can be made to ignore certain "hits"	No	Partial -- can be made to ignore certain identifiers	No
4.1.2	Prioritize Data and Data Reduction	No	No	Partial	No	No	Yes	Yes
4.1.3	Data Visualization	No	No	No	No	No	No	No
4.1.4	Import and Export Formats that are supported	No	No		Yes - Hitlists can be saved.	No	No	Yes -- saved as a history file
4.1.5	Differential Reporting and Trend Analysis	No	No	No	Yes	No	No	Partial -- Jint will not repeat reporting of results previously discovered
4.1.6	Report generation templates, creation and customization of templates, and fine tuning of report outputs	No	Partial	No	No	No	Partial	No
4.1.7	Auto Fix or Fix Information	No	No	No	No	No	Yes -- Some solution information can be provided	No
<b>4.2 Security</b>								
4.2.1	Access to the tool	No	No	No	No	No	No	No
4.2.2	Logging of Users and their actions	No	No	No	No	No	No	No
4.2.3	Tampering and Integrity Control	No	No	No	No	No	No	No
4.2.4	Tools or Users access to targets	No	No	No	No	No	No	No

**Table 3-3 Developer (Application Security Assessment Tools)**

PyChecker	RATS	Splint
PyChecker (0.8.11)	RATS (Rough Auditing Tool for Security) (v 1.5)	Splint (Secure Programming Lint)
<a href="http://pychecker.sourceforge.net/">http://pychecker.sourceforge.net/</a>	Secure Software ( <a href="http://www.securesw.com/rats.php">http://www.securesw.com/rats.php</a> )	University of Virginia ( <a href="http://www.splint.org/">http://www.splint.org/</a> )
Free	GNU General Public License -- Open Source	GNU General Public License -- Open Source
	UNIX, Windows (Windows version requires Expat from <a href="http://sourceforge.com">sourceforge.com</a> )	UNIX, Linux, FreeBSD, Solaris, Win32, OS/2
Python Source Code	C, C++, Python, Perl and PHP Source Code	C Source Code
No	No	No
	Makefile included	Makefile included
Simple GUI	Command-line driven	Command-line driven
Simple GUI and Command-line driven	No	Command-line driven. Can be integrated into the MS Visual Studio IDE
No	No	No
No	No	No
	Automated	
n/a	n/a	n/a
None	None	None
via web-site	via web-site	via web-site
via web-site	via web-site	via web-site
Minimal	Minimal	Complete
n/a	No	n/a
No	Vulnerability saved as XML files which can be modified.	
No	No	No

**FOR INFORMATION PURPOSE**

**Table 3-3 Developer (Application Security Assessment Tools)**

Command-line driven	Command-line driven	Command-line driven
No	Partial	Yes, via code annotation
No	No	
Yes	Yes	Yes
Yes -- Python code	Yes -- C, C++, Perl, PHP and Python code	Yes -- C code
No	No	No
Partial	Partial	Partial
No	Partial	Partial
No	No	No
No	No	No
No	No	No
Partial -- Certain warnings can be suppressed	No	Yes
No	Partial	
No	No	No

### **3.4 GENERAL-PURPOSE ASAT TABLE**

Table 3-4 presents general-purpose ASATs identified while conducting the market survey. The table contains a mapping of each tool's capabilities to the capability definitions presented in Section 2.2.

*[Note: For convenience, the General-Purpose Tool worksheet has been pasted in the pages below. To view the entire worksheet with survey notes that correspond to specific cells, you will need to refer to the Microsoft Excel file (Market Survey Matrix 72502.xls) and the worksheet labeled "General Purpose."]*

**Draft**  
**Table 3-4 General Purpose (Application Security Assessment Tools)**

Number	Criteria	BV-Control	Foundscan	Nessus	NetRecon	Retina	SAINT	Security Analyzer	STAT Scanner	Typhon II
<b>1 General Information</b>										
1.1	<b>Name / Version</b>	Bv-Control	Foundscan 2.5	Nessus Security Scanner	NetRecon 3.5	Retina	SAINT	Security Analyzer	STAT Scanner	Typhon II
1.2	<b>Vendor or Source of Tool</b>	BindView	Foundstone Inc.	Nessus	Symantec	eEye Digital Security	Saint Corporation	NetIQ	Harris	Next Generation Software
1.3	<b>Cost</b>	Unknown	\$30,000+ plus variable maintenance fees	Free (Open-Source)	Unknown	Variable based on IPs licensed, See Comment	Free (Open-Source), with purchase of SAINTwriter or SAINTexpress	Unknown	Variable based on IPs licensed, See Comment	Enterprise or Consultant Version, License Valid for 1 Year
1.4	<b>Scanner Platform/Architecture</b>	See Comment	Dedicated Database Server and Web Portal Server	Client-Server	Windows NT 4 SP 3+ or Windows 2000 Pro/Server	Windows NT/2000/XP	UNIX Based, See Comment	See Comment	See Comment	Windows NT/2000
1.5	<b>Scanning Targets</b>	See Comment	Network Devices, by IP	Network Devices	Unix, Linux, Windows 95/98/NT/2000, NetWare, Network Devices	Any OS's or Network Devices	Remote Hosts and Networks	See Comment	See Comment	Windows NT/2000
1.6	<b>Assurance</b>	Yes, See Comment	No Known Certification	No Known Certification	No Known Certification	No Known Certification	No Known Certifications	No Known Certification	No Known Certification	No Known Certification
<b>2 Usability Attributes</b>										
<b>2.1 Ease of Use</b>										
2.1.1	Installation, Updates, and Maintenance	Simple	Multiple Computers with Multiple Application required	Moderately Difficult	Simple	Simple	Slightly Difficult to Difficult depending on technical level	Simple	Simple	Simple
2.1.2	Configuration	Simple Configuration, GUI	Simple Configuration, GUI	Simple Configuration, GUI	Simple Configuration, GUI	Simple Configuration, GUI	Difficult for Novice	Simple Configuration, GUI	Simple Configuration, GUI	Simple Configuration, GUI
2.1.3	Intuitive GUI	Yes	Yes	Yes	Yes	Yes	Deceptive	Yes	Yes	Yes
2.1.4	Crash Recovery	No	No	No	No	No	Yes	No	No	No
2.1.5	Emergency STOP	Yes	Unknown	Yes	Yes	Yes	Yes	Yes	Yes	Yes
2.1.6	Methods for running tasks	Manual, Automatic	Manual, Automatic, Continuous	Manual	Automatic, Manual	Manual, Schedule	Manual, Scheduled	Manual, Scheduled	Manual	Manual
2.1.7	Ability to save profiles or sessions	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
2.1.8	Potential to cause damage limited or clearly defined	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
<b>2.2 Support</b>										
2.2.1	Type of Support	web, email, and phone dependent upon severity	24x7 with maintenance contract	Limited	Web, Email, Phone	web, email, phone	Limited	Extra, phone and email, premium plan includes 24x7 support	Available, type unknown	Email
2.2.2	Updates	Frequent, automatically by secure broadcast	Frequent, often same day as vulnerabilities found	Almost Daily, updates freely available through HTTP, FTP	LiveUpdate Capability	Nearly Daily, automatic update or manual	Yes, Frequent	On-Demand access to most recent security tests available	Monthly	Unknown, Manual check for updates button though
2.2.3	Bug Fixes and Client Input	Yes	Yes	Yes, Email	Yes	Yes	Yes, Email	Yes	Yes	Email
<b>2.3 Documentation</b>										
2.3.1	Complete	Unknown	Unknown	Yes	No Manuals	Yes	Yes	Yes	Yes	Yes
2.3.2	Updated in a Timely Manner	Unknown	Unknown	Yes	No	Unknown	Yes	Yes	Unknown	Unknown
<b>3 Technical Attributes</b>										
<b>3.1 Configuration Customization</b>										
Allows the use of scripting to customize the application, its program modules or scan tests										
3.1.1		No	Proprietary FASL scripting language, unknown is available to user	Yes	No	Yes	Yes	Yes	No	No
3.1.3	More than one scripting language can be used	No	No, only custom proprietary FASL scripting	Yes	No	Yes	Yes	No	No	No
3.1.4	Can be tailored for multiple environment configurations of scan and vulnerability settings.	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
3.1.5	Supports fine-tuning of scanner (e.g., minimize false-positive/false-negative conditions).	Unknown	Yes	Yes	No	Yes	Yes	Yes	Not Exactly	No
3.1.6	Provides a capability to modify system scan profiles, including multiple criteria.	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No
<b>3.2 Vulnerabilities Scanned</b>										

**Draft**  
**Table 3-4 General Purpose (Application Security Assessment Tools)**

3.2.1	Check for known vulnerabilities associated with multiple operating systems.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
3.2.2	Check for known vulnerabilities associated with various applications.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
3.2.3	Check for known vulnerabilities associated with web server vulnerabilities.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
3.2.4	Check for known vulnerabilities associated with mobile code.	Unknown	Unknown	Unknown	Unknown	Unknown	No	Unknown	Unknown	Unknown	
3.2.6	Check for known vulnerabilities associated with Browsers.	Unknown	Unknown	Unknown	Yes	Unknown	No	Yes	Yes	Yes	
3.2.7	Check for vulnerable software programs.	Unknown	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Limited	
3.2.8	Checks for network device configurations vulnerabilities.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
3.2.10	Checks for known vulnerabilities associated with network services/protocols.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
3.2.11	Checks for the SANS Top Vulnerabilities	Yes	No	Unknown	No	No	Yes	Yes	No	No	
3.2.13	Capable of discovering vulnerabilities not previously identified.										Yes
3.2.13.1	Cookie Poisoning	No	No	No	No	No	No	No	No	No	
3.2.13.2	Hidden Field Manipulation	No	No	No	No	No	No	No	No	No	
3.2.13.3	Parameter Tampering	No	Yes	No	No	No	No	No	No	No	
3.2.13.4	Buffer Overflow Attacks	No	No	No	No	Yes	No	No	No	No	
3.2.13.5	Cross-Site Scripting	No	No	No	No	No	No	No	No	No	
3.2.13.6	Backdoor and Debug Options	No	No	No	No	No	No	No	No	No	
3.2.13.7	Forceful Browsing	No	No	No	No	No	No	No	No	No	
3.2.13.8	Stealth Commanding	No	No	No	No	No	No	No	No	No	
3.2.13.9	3rd Party Misconfiguration	No	Yes	No	No	No	No	No	No	No	
3.2.13.10	Database Sabotage (SQL Injection)	No	Yes	No	No	No	No	No	No	No	
3.2.13.11	Data Encoding	No	No	No	No	No	No	No	No	No	
3.2.13.12	Protocol Piggyback	No	No	No	No	No	No	No	No	No	
3.2.14	Capable of discovering potential vulnerabilities in Source Code										No
<b>3.3</b>	<b>Requirements Scanned (Policy and Procedure Compliance)</b>										
3.3.1	Checks passwords for strength, complexity, and compliance with established security policy.	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes	
3.3.2	Checks for unusual account activity.	No	No	No	No	No	No	No	No	No	
3.3.3	Identifies the groups and accounts associated with those groups.	Yes	No	Yes	Yes	No	No	Yes	No	Yes	
3.3.4	Checks account privileges.	Yes	No	Yes	Yes	No	No	Yes	No	Yes	
3.3.5	Identifies inactive accounts.	Yes	No	Yes	Yes	No	No	Yes	Yes	Yes	
3.3.6	Check of system auditing.	Yes	No	No	Yes	No	No	Yes	Yes	Yes	

**FOR INFORMATION PURPOSE**

**Draft**  
**Table 3-4 General Purpose (Application Security Assessment Tools)**

3.3.7	Check for key lengths in browsers and other applications.	No	No	No	No	No	No	No	No	No
<b>4</b>	<b>Operational Attributes</b>									
<b>4.1</b>	<b>Reporting</b>									
4.1.1	Parse/Normalize/Query/Analytical Functions	Yes	No	No	Yes	No	No	No	No	No
4.1.2	Prioritize Data and Data Reduction	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	No
4.1.3	Data Visualization	Yes	Yes	Yes	Yes	Yes	If purchase SAINTwriter	Yes	Yes	No
4.1.4	Import and Export Formats that are supported	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
4.1.5	Differential Reporting and Trend Analysis	Differential	Yes	No	No	No	No	Yes, Comparative Reports	Differential	No
4.1.6	Report generation templates, creation and customization of templates, and fine tuning of report outputs	Yes	No	No	Yes	Yes	No	Yes	No	No
4.1.8	Auto Fix or Fix Information	Auto Fix and Fix Information	Fix Information	Fix Information	Fix Information	Auto Fix and Fix Information	Fix Information	Fix Information	Auto Fix and Fix Information	Fix Information
<b>4.2</b>	<b>Security</b>									
4.2.1	Access to the tool	No	Yes	Yes	No	No	Root	No	Yes	No
4.2.2	Logging of Users and their actions	No	No	No	No	No	No	No	No	No
4.2.3	Tampering and Integrity Control	No	No	No	No	No	No	No	No	No
4.2.4	Tools or Users access to targets	By License	No	No	License	By License	Yes, at users control	License	Yes	License

## 4. REVIEWS

### 4.1 WEB APPLICATION ASATs

#### Overview:

Web application vulnerability assessment tools are designed to automatically scan Web applications, sites, and servers, looking for potential vulnerabilities. These tools differ from general vulnerability assessment tools (see Section 4.4) in that they do not perform a broad range of checks on a myriad of software and hardware (i.e., port scanning or host vulnerability scanning). Instead, they perform other checks, such as potential field manipulation and cookie poisoning, which allows a more focused assessment of Web applications by exposing vulnerabilities that standard VA tools are unaware of.

The following tools are reviewed in this section:

- AppScan
- DominoScan
- HailStorm
- N-Stealth
- Nikto
- ScanDo
- WebEnforcer
- WebInspect
- Web Scarab
- WebSleuth
- whisker (and libwhisker)
- WhiteHat Arsenal

#### Reviews:

##### 4.1.1 AppScan

Product	<b>AppScan</b>
Vendor	Sanctum, Inc.
URL	<a href="http://www.sanctuminc.com">www.sanctuminc.com</a>

#### Product Description:

AppScan is a scanner designed to perform automated security risk assessments of Web-based applications to determine weaknesses that could be exploited by hackers. AppScan analyzes the behavior of the actual Web application and the vulnerabilities it finds, exposing security loopholes (such as parameter tampering, forceful browsing, cross-site scripting, and hidden field manipulation) that occur in the application code and within widely used third-party products. AppScan dynamically scans the application by analyzing the outbound Hyper Text Markup Language (HTML) pages on the fly as they will be seen by the legitimate user and hacker. During an explore stage, AppScan automatically visits every page of a site, except for those filtered by configuration settings, and analyzes the applications handling of the Hyper Text Transfer Protocol (HTTP) requests and responses. In the process, it detects potential vulnerabilities in the forms, HTML code, and CGIs. Each link on the site is analyzed and recorded in the products database. The user can define which types of attacks to execute and whether to

## Draft

perform them automatically or manually. Using input from its expert system, AppScan automatically assigns severity and success ratings for tested attacks and provides guidance for fixing the vulnerabilities. AppScan automatically generates preconfigured reports in both textual and graphical format, and can be customized to reflect the expertise and information needs of the user-high-level analysis for the executive summary, technical details for security experts, and recommended code fixes for Quality Assurance (QA) and developers.

### Comments:

AppScan is a powerful and fast vulnerability scanner, performing thousands of security checks quickly. The product is designed to uncover difficult-to-detect and unknown vulnerabilities in a Web application code, including those caused by buffer overflows, cross-site scripting, parameter tampering, and forceful browsing among others. AppScan presents the user with an intuitive GUI that follows a clear scanning, assessment, and reporting methodology. Although AppScan allows some customization in the scans that it performs (i.e., automated, interactive, or user-defined), the product does not allow users to develop their own tests using scripts. AppScan does, however, allow users to manipulate the parameters of built-in tests. Although AppScan offers a broad reporting capability and extensive guidance on mitigating any found vulnerabilities, it does not directly provide an ability to perform trend analysis by comparing previous scans. AppScan's \$15,000 cost (excluding a variable annual maintenance fee) allows scanning of all the domain names and Internet Protocol (IP) addresses owned by the purchaser.

### 4.1.2 DominoScan

Product	<b>DominoScan 1.1</b>
Vendor	Next-Generation Software
URL	<a href="http://www.nextgenss.com/dominoscan.html">http://www.nextgenss.com/dominoscan.html</a>

### Product Description:

DominoScan is a specific application scanner designed for Lotus Domino Web Servers. This tool allows a user to not only test the Web Server for vulnerabilities or weak spots, but also test and assess all applications running on the Domino Server that were written in-house. Each document found is then put through a series of tests that demonstrates what an attacker can do and gain access to.

There are three aspects to the auditing procedure of DominoScan. First, it will examine the server for known security vulnerabilities in the Web server itself. Second, DominoScan will attempt to gain access to more than 100 sensitive and or default databases such as names and log.nsf. Third, custom-created databases will be audited. In this part of the audit, DominoScan will discover every view, form, agent, and document and will put each through a vigorous set of checks to ascertain risk exposure. Some of the features of DominoScan are as follows:

- Attempts to gain access to more than 100 sensitive/default databases
- Web Administrator template access using ReplicaID and buffer truncation
- cache.dsk access using buffer truncation

**FOR INFORMATION PURPOSE**

## Draft

- Directory traversal
- Database browsing
- Audits custom-built databases
- Unique database structure enumeration technology
- Finds hidden and visible views
- Default Navigator Access
- Attempts to bypass Default Navigator protection
- Evaluates Database Design
- Checks every document for Edit access
- Attempts a forced search
- ReadEntries and ReadViewEntries access
- Detailed post-audit HTML report generated
- Easy to use and highly configurable
- Performs focused audits
- Scans as an authenticated user
- Performs QuickHit audit
- Very fast and cost effective.

### Comments:

Dominoscan contains a straightforward, easy-to-use Graphical User Interface (GUI). The tool can also be run from the command prompt so as to allow the scheduling of periodic scans via a scheduler service or through the use of batch files to scan multiple hosts. Note, however, that the tool does not allow much configuration of the options or tests to be performed. This tool is programmed to run its tests in a quick manner; therefore, test configuration may not be necessary or desired.

### 4.1.3 HailStorm

Product	<b>Hailstorm</b>
Vendor	Cenzic
URL	<a href="http://www.cenzic.com/">http://www.cenzic.com/</a>

### Product Description:

HailStorm is a tool for scanning any network device, hardware, or application. The scanning that is performed is for both known and unknown, or undiscovered, vulnerabilities. In addition to scanning and testing for these known vulnerabilities and classes of unknown vulnerabilities, HailStorm also provides the capability and foundation for creating custom tests.

At the core of the testing process, HailStorm's methodology of fault injection is used to inject controlled faults into a system so as to expose and analyze real and potential failures at all levels. HailStorm automates the manual security testing process using fault injection to subject both the network and applications to controlled hostile scenarios while observing the failures. It then provides analysis to identify fault location, category, and the actual transaction that delivered the failure. Finally, it will assign a degree of risk

**FOR INFORMATION PURPOSE**

## Draft

to each of these failures in addition to architecture and implementation strategy in mitigating this risk level.

HailStorm provides three levels of user interface: Novice, Intermediate, and Expert. These levels range from full automation of testing for non-security personnel, to drag-and-drop testing with customization for QA testing, security administrators, and consultants, to full customization of scans for advanced security professionals.

HailStorm's fault injection tests are as follows:

- Buffer Overflow
- Format String Buffer Overflows
- Command Injection
- SQL Database Attacks
- Parser and Metacharacter Injection
- Invalid Application States
- Information Leakage
- Denial of Service (DOS) and Distributed Denial of Service (DdoS)
- Improper Configuration
- Client-Side Content Attacks (CSS)
- Cross-Site Scripting
- Filter and Firewall Evasion
- Intrusion Detection System (IDS) Evasion
- Character Encoding
- String Construction
- File and System Calls
- Unfiltered User Input
- Transmission Control Protocol (TCP) Segmentation
- IP Fragmentation

### Comments:

Although HailStorm is able to discover known vulnerabilities, its main strength is as an extremely powerful QA testing tool for discovering unknown and potential vulnerabilities. Although wizards and automated methods exist for testing, the full power of the tool will not be used without Expert users and a thorough study of the tool. As unique potential vulnerabilities are discovered in the Expert testing mode, automated tests can then be created for those discovered problem areas for future Novice or Intermediate user level testing. This is an extremely versatile tool with no limit on the type, size, and variation of network traffic it is capable of creating in its very fast attempts to inject faults into network applications. Lastly, note that HailStorm is quite costly (\$30,000), which may be prohibitive to organizations deploying this tool in large numbers.

### 4.1.4 N-Stealth

Product	<b>N-Stealth</b>
Vendor	N-Stalker
URL	<a href="http://www.nstalker.com/nstealth/">http://www.nstalker.com/nstealth/</a>

### Product Description:

N-Stealth is a Web vulnerability scanner that assesses Web servers to identify known and unknown vulnerabilities, security problems, and weaknesses that may be exploitable. This tool checks for more than 19,000 vulnerabilities, including checks that cover the

**FOR INFORMATION PURPOSE**

## Draft

Top 20 SANS/FBI vulnerabilities. Options exist for a complete scan of all possible vulnerabilities, a “normal” reduced scan for a reduced set of vulnerabilities, and vulnerabilities that correspond to either the SANS/FBI Top 10 or Top 20 list. Additional changes to the vulnerabilities scanned for is limited by category selection or the manual editing of a configuration file.

N-Stealth contains filters to help reduce false positives while maintaining complete and accurate reporting that includes Bugtraq and CVE-compatible vulnerability information. A buffer overflow engine assists in detecting of unknown vulnerabilities. User defined vulnerability checks may also be added easily via configuration files accessible within N-Stealth.

### Comments:

Although N-Stealth can be a powerful tool in identifying known and possible unknown vulnerabilities, it requires some effort to become proficient in its use. The GUI is neither intuitive nor easy to navigate, thereby complicating the learning process. Because of the nature of some of the specific tests that are performed, the information regarding the potential vulnerabilities discovered is not always helpful or extensive, leaving additional follow-up work for the user. This tool and its generated reports are more appropriate for use by those with a high level of technical background.

### 4.1.5 Nikto

Product	<b>Nikto</b>
Vendor	Chris Sullo (Open Source)
URL	<a href="http://www.cirt.net/code/nikto.shtml">www.cirt.net/code/nikto.shtml</a>

### Product Description:

Nikto is a Web server scanner that performs comprehensive tests against Web servers. Nikto is designed to examine Web servers and search for items in multiple categories: misconfigurations, default files and scripts, insecure files and scripts, and outdated software. Nikto’s checks, both information and actual security problems, are derived from a number of sources. These include the mailing lists (BugTraq, NTBugTraq, WebAppSec [WWW-Mobile-Code]), and the standard Web sites: [www.securitytracker.com](http://www.securitytracker.com), [www.securiteam.com](http://www.securiteam.com), [www.packetstormsecurity.com](http://www.packetstormsecurity.com), and [www.securityfocus.com](http://www.securityfocus.com)). In addition, the developer monitors updates to Nessus project.

Nikto is a Perl script that requires that the user have some version of that language installed. This is an important consideration in that the tool can be manually updated by the user coding a custom-built plug-in.

The tool uses Rain Forest Puppys LibWhisker (see Whisker below) for HTTP (network) functionality, and can perform checks in HTTP or Hyper Text Transfer Protocol Secure (HTTPS). It also supports basic port scanning and will determine if a Web server is running on any open ports. Although the LibWhisker library is included with Nikto, it is

## Draft

recommended that the user download the latest version directly from the wiretrip.net Website.

### Comments:

Nikto is a fast scanner and some of its settings can generate more than 70,000 HTTP requests to a target. As a result, its use can be potentially damaging to a Web site. Because Nikto is closely tied to many Web-based vulnerability projects and is open source, the tool will remain an up-to-date and easy-to-custom program, even if its solitary developer stops supporting the tool. However, because the tool is a command-line driven Perl script, it is not inherently user friendly, which may dissuade some potential users.

### 4.1.6 ScanDo

Product	<b>ScanDo</b>
Vendor	Kavado
URL	<a href="http://www.kavado.com/ProductsScando.htm">http://www.kavado.com/ProductsScando.htm</a>

### Product Description:

ScanDo is a tool designed to scan an entire Web application while registering all of its structure and content. During this process, ScanDo will assess the entire Web application to determine its susceptibility to security breaches. After this assessment period, ScanDo has the option to attack each vulnerability that is found to verify and further test the Web application. These findings can then be used to narrow down the possible false positives.

In assessing and attacking a Web application, ScanDo uses multiple techniques implementing each as separate components. ScanDo also provides support for custom scripts using VB script or Jscript. In addition, the ScanDo application programming interfaces (API) are exposed to support the user in writing their own attack and assessment methods.

ScanDo not only uncovers known vulnerabilities, but also provides methods for discovering an array of unknown possible vulnerabilities, including the following:

- Manipulation of Information Technology (IT) infrastructure vulnerabilities
- Parameters tampering and hidden fields manipulation
- Cookie poisoning
- Stealth commanding
- Backdoor and debug options
- Third-party misconfiguration
- Database sabotage
- Buffer overflow attacks
- Data encoding
- Protocol piggyback.

ScanDo is able to automatically generate reports for the management level and technical user. These reports can also be customized to suit individual requirements. All

**FOR INFORMATION PURPOSE**

## Draft

assessment and attack data may also be exported for external uses in Extensible Markup Language (XML) or Assign Digital Transmission Group (ADTG) formats.

### Comments:

ScanDo has a wizard-driven interface that is easy to learn even the difficult assessments for the unknown vulnerabilities. There is also support for custom attack scripts along with a manual attack utility, so that all types of attacks can be performed against Web applications. ScanDo can also be updated with new hacking methods in the event new methods need to be included to automate testing for unknown vulnerabilities. Not only is this automated assessment process simple to perform, but also each potential vulnerability can either be noted, or ScanDo can attempt to actually exploit the potential vulnerability for further confirmation that a new unknown vulnerability exists. Overall, ScanDo is an easy-to-use tool that remains powerful and full of capabilities.

### 4.1.7 WebEnforcer

Product	<b>WebEnforcer 2.0</b>
Vendor	Hewlett Packard
URL	<a href="http://www.hp.com/security/products/webenforcer/">http://www.hp.com/security/products/webenforcer/</a>

### Product Description:

WebEnforcer is a Windows NT/2000 Web security tool used for testing key components of the Windows Web Server Environment. These components are as follows:

- Windows Server
- IIS Web Server
- Transaction Server
- Index Server
- Internet Explorer (IE)
- Data Access Components.

WebEnforcer is designed to not only test and repair hundreds of vulnerabilities, but also continuously monitor and enforce security policies. For those vulnerabilities that the tool cannot automatically fix, information is provided for correction of the vulnerability manually. For an extra fee the HP SecurityUpdate subscription service ensures that WebEnforcer protection incorporates recently discovered security threats and attacks.

### Comments:

WebEnforcer contains a straightforward, easy-to-use GUI. This interface not only enables a user to scan for vulnerabilities, but also allows the product to monitor and maintain a secure configuration. A wizard interface allows for the creation of profiles for scanning. The reporting features provide detailed information, but do not appear to allow any customization.

### 4.1.8 WebInspect

Product	<b>WebInspect</b>
---------	-------------------

**FOR INFORMATION PURPOSE**

**Draft**

Vendor	SPI Dynamics
URL	<a href="http://www.spidynamics.com">www.spidynamics.com</a>

**Product Description:**

WebInspect is designed to dynamically scan standard and proprietary Web applications to identify known and unknown application vulnerabilities. WebInspect deploys assessment agents to examine applications or servers and decide which threat agents to deploy. The threat agents then vet applications against known software vulnerabilities and try to automatically hack into an application to discover flaws. WebInspect uses “Adaptive-Agent” technology, a set of heuristics that enable it to apply intelligent application-level security checks. This technology is a multiphase approach to Web application assessments. As a user initiates an assessment, WebInspect assigns assessment agents to dynamically catalog all areas of a Web application. As these agents complete the assessment, findings are reported back to a main security engine that analyzes the results. WebInspect then launches “threat agents” to evaluate the gathered information and apply attack algorithms to determine what vulnerabilities exist and the severity of those vulnerabilities. WebInspect provides a full programming language and programming tools to write custom rules. WebInspect electronically updates its built-in intelligence as it references and synchronizes with a continuously updated database of hacking methodologies over the Internet.

**Comments:**

WebInspect is a powerful and flexible application vulnerability scanner. It allows the user to conduct security assessments on any Web-enabled application, including specific assessment capabilities for IBM WebSphere, Lotus Domino, Oracle Application Servers, and MacroMedia ColdFusion. Configuring the system to analyze Internet applications is also quickly and easily accomplished, although this can vary depending on the number of systems to be analyzed. The tool provides advanced executive-level reporting functionality with additional report and graphing features, including trend analysis, for comparing assessments and tracking progress. WebInspect costs \$4,995 per physical server scanned.

**4.1.9 Web Scarab**

Product	<b>Web Scarab</b>
Vendor	Open Web Application Security Project (OWASP)
URL	<a href="http://www.owasp.org">www.owasp.org</a>

**Product Description:**

Web Scarab is a project (still under development) that will be designed to build a true Open Source Web application security assessment tool. The tool will be able to examine a complete Web site or individual applications running within a Web site for security issues. Web Scarab will focus on only Web application security issues. It will not duplicate Web server checks and operating system (OS) checks found in scanners like Nessus (although it may import them). It will be capable of allowing users to enter their own checks as well as being automated. It will guide a user conducting structured

## Draft

security testing and support the OWASP testing framework being developed. The tool will be platform independent (probably written in Java) and 100 percent open source using an approved Open Source license. It will be designed in such a way that novices and highly skilled technicians can use its various features, and it will be driven automatically by machine scripts (or scheduling). The tool will be capable of exposing the raw underlying workings and hiding them. It will be extensible, allowing users to add their own custom checks either in Java or via an intermediary language or XML. The tool will be able to check for static vulnerabilities and dynamic vulnerabilities. The tool will use WebSphinx, a Web-spider written in Java.

### Comments:

The Web Scarab project is still in its early stages of development. As of early July 2002, the project had been outlined but lacked any developed code. However, even in its nascent stage, Web Scarab offers tremendous promise in its proposed capabilities. Offering both static and dynamic tests, Web Scarab will examine a broad range of vulnerabilities, including SQL injection, buffer overflows, meta characters, null characters, and directory traversal. Moreover, because of its open source nature, Web Scarab's code will be open to scrutiny and customization. Although no firm schedule is provided development and release, this will be a tool to watch out for in the future.

### 4.1.10 WebSleuth

Product	<b>WebSleuth</b>
Vendor	Open Source (see URL below)
URL	<a href="http://www.geocities.com/dzzie/sleuth/">www.geocities.com/dzzie/sleuth/</a>

### Product Description:

Although WebSleuth is listed as a Web application scanner, this is somewhat a misnomer. Although WebSleuth does allow a user to test a Web site and application for vulnerabilities, it does not do so automatically for the user. Instead, WebSleuth provides a proxy between users and the Web, allowing them to manipulate and customize their interaction with the Web and subsequently analyzing the results. As the developer puts it on the tools homepage, WebSleuth is "a powerful efficient, intuitive, technical tool. It may take more knowledge and ability to operate and make sense of returned results...but such are the tradeoffs." WebSleuth provides the user with a simple GUI to perform the following:

- Convert hidden and select form elements to textboxes
- Parse and analyze form fields
- Edit rendered Web page source code
- Interface with edit cookies windows API
- Make raw HTTP requests to servers to fake elements such as referrer and cookie.
- Provide automatic template source filtering
- Provide HTML source code highlighting and parsing
- Analyze CGI links and prompt before navigation
- Log all surfing activities
- Generate reports of elements of Web page.

**FOR INFORMATION PURPOSE**

## Draft

In addition, WebSleuth supports third-party plugins to add functionality as a result of its open source nature. Plugins have been developed to crack session-ids in cookies and uniform resource locators (URL) and basic authentication using brute force techniques; to perform Structured Query Language (SQL) injection; to test form inputs for cross-site scripting holes and SQL injection; and to perform server file enumeration.

### Comments:

WebSleuth clearly is a powerful tool, but one that requires some expertise in its usage. WebSleuth provides a GUI to simplify tests against Web sites while searching for vulnerabilities. However, the logic and intuition behind the tests is provided solely by the user; WebSleuth provides a mechanism for performing the tests with more ease. WebSleuth, which is open source, provides a means for extensive customization by third-party developers. As such, it may prove to be a very useful tool when analyzing custom sites and applications for vulnerabilities.

#### 4.1.11 whisker (and libwhisker)

Product	<b>whisker and libwhisker</b>
Vendor	Rain Forrest Puppy (Open Source)
URL	<a href="http://www.wiretrip.net">www.wiretrip.net</a>

### Product Description:

Whisker is a Web Common Gateway Interface (CGI) scanning tool written by Rain Forrest Puppy that has been in existence for some time. Whisker attempts to logically search for CGI programs versus a brute-force URL search approach. Libwhisker is a programming API used by whisker v2.0. It exists separately from whisker v2.0 and can be used by anyone wishing to perform various Web/HTTP functions. whisker's documentation explains its simple script database language that is the key to driving the tool. The language has numerous capabilities (such as if/then logic, internal variables) that can be effectively employed to extend whisker's capabilities.

Libwhisker, which is a Perl module geared specifically for HTTP testing, has a few design principles:

- Portable - runs with 0 changes on UNIX, Windows, etc.
- Flexible - was designed with a no rules approach
- Contained - does not require any external Perl modules
- Localized - does not require installation to use.

Note that libwhisker is not an executable program, but merely a library that provides the following types of capabilities to a program:

- Allows the program to communicate over HTTP 0.9, 1.0, and 1.1
- Uses persistent connections
- Has proxy support
- Has anti-IDS support
- Has Secure Socket Layer (SSL) support
- Handles chunked encoding

**FOR INFORMATION PURPOSE**

## Draft

- Has nonblock/timeout support built in (platform-dependent).

### Comments:

Whisker is a useful CGI tool, but is in need of updating. According to its creator, who recently released a much-anticipated v2.0 of the tool, v2.1 will be the first “new” general release of the program that will be documented and supported. Although whisker’s current version deserves considerable merit, a few items should be noted. Although whisker employs a quick-to-learn scripting language, it is clearly a tool meant for the expert user. Note that this tool is run from the command line and requires a significant level of expertise to unleash its full capabilities with some degree of effectiveness. Also, today’s Web environment has extended beyond the CGI and is considerably more complex, effectively limiting whisker’s utility. On the other hand, libwhisker is a tremendously useful library employed by many programs (such as Nikto and WhiteHat Arsenal). It is fairly well documented and can be included and used within a program without much difficulty.

### 4.1.12 WhiteHat Arsenal

Product	WhiteHat Arsenal
Vendor	Community WhiteHat Security
URL	<a href="http://community.whitehatsec.com/wharsenal/">http://community.whitehatsec.com/wharsenal/</a>

### Product Description:

WhiteHat Arsenal is designed from the ground up to be a generic Web application security productivity tool and possesses a powerful suite of GUI browser-based Web security tools. The open source tool focuses attention on HTML forms, permitting the user to view form inputs, including hidden fields, and easily modifies them, allowing the tool to uncover vulnerabilities in a Web application. WhiteHat Arsenal mimics the browser (mimicking the HTTP Request behavior of a standard Web browser) and as a proxy allowing the user to traverse Web sites while modifying HTTP requests. The tool operates in one of three modes: spider, ripper, or forced browsing. The spider mode takes a Web server target and traverses the site completely, showing all the pages visited page information. The ripper mode allows the user to edit HTML forms on-the-fly and provides advanced control over HTTP requests such as viewing and editing the request/response header. The forced browsing mode allows the user to find hidden directories, log files, and backup files that may contain useful information quickly, easily and efficiently. Ultimately, WhiteHat Arsenal provides an ability to modify and manipulate most aspects of an HTTP request (i.e., path, protocol, port, content, and method). WhiteHat Arsenal logs all HTTP request activities in either XML or HTML format.

### Comments:

WhiteHat Arsenal is a powerful tool for assisting in the search for Web site vulnerabilities, but one that requires some expertise in its usage. WhiteHat Arsenal does provide a GUI to simplify tests against Web site, but the user must provide the logic and expertise needed to interpret the results. The tool does provide good logging capabilities to record all HTTP request activities, is open source, and is updated frequently.

**FOR INFORMATION PURPOSE**

## 4.2 DATABASE ASATS

### Overview:

Database vulnerability assessment tools are hybrid scanners designed to specifically evaluate and assess database vulnerabilities. These tools perform penetration testing and auditing, generally scanning for known configuration vulnerabilities, incorrect settings, weak security profiles, and missing patches/out-of-date software.

The following tools are reviewed in this section:

- AppDetective
- Database Scanner
- OraScan.

### Reviews:

#### 4.2.1 AppDetective

Product	<b>AppDetective for Oracle</b>
Vendor	Application Security Inc.
URL	<a href="http://www.appsecinc.com">www.appsecinc.com</a>

### Product Description

AppDetective is a network-based, penetration testing/vulnerability assessment scanner that locates and assesses the security strength of database and groupware applications within a network. AppDetective will locate, examine, report, and help fix security holes and misconfigurations. AppDetective systematically scans a network for database and database components, providing version numbers and names of each discovered component. AppDetective then performs a series of tests to identify how an intruder or unauthorized user could gain access to the system using an “outside-in” approach to security. In addition, a security audit can be performed to provide an in-depth examination of the internal configurations and potential security holes within a database. This “inside-out” approach requires access to the database as a valid user to verify internal configuration settings. A security audit also examines how an unauthorized user can obtain elevated privileges or circumvent security controls and mechanisms of a database from the inside. Versions of AppDetective also are available for SQL Server, Sybase, and Lotus Domino.

### Comments:

AppDetective is a robust database vulnerability assessment tool, providing inside-out configuration scans, and outside-in penetration testing. In this capacity, AppDetective may prove to be a useful tool to verify the database requirements as established by the *Application Security Requirements Guide*. For example, AppDetective assists in identifying the following types of vulnerabilities:

- Access control
- Application integrity
- Identification/password control

**FOR INFORMATION PURPOSE**

- OS integrity.

AppDetective uses a simple GUI to automate the scanning and auditing process, and its results are presented in a clear manner.

#### 4.2.2 Database Scanner

Product	Database Scanner
Vendor	Internet Security Systems
URL	<a href="http://www.iss.net">www.iss.net</a>

##### Product Description:

The Database Scanner application is a stand-alone application that identifies security exposures in leading database applications. Database Scanner offers security policy generation and reporting functionality, which measures policy compliance and automates the process of securing critical online data. Database Scanner detects weak passwords, checks password aging (expiration), detects login attacks, disables stale logins (old unused accounts), and tracks login hour restrictions. Database Scanner can perform two types of database scans: an audit scan and a penetration test. Auditing is an inside-out approach that enumerates users, groups, privileges, logins, and a wide number of other objects in the database, identifying misconfigured privileges and opportunities for misuse by authorized users. Database Scanner’s auditing feature allows users to know exactly what objects are in their database, who has access to them, and what they have been doing and could do. Penetration testing, on the other hand, is an “outside-in” approach that attempts to gain access to a database the way a hacker would by using known default passwords and password guessing. Database Scanner 4.2 includes a set of predefined, customizable “best practice” security policies that allows test ranging from analyzing the risk of compromise from simple attacks from unsophisticated external attackers to testing the integrity of application data and customer-specific application configurations against accidental or malicious changes. Database Scanner also allows the customization of policy templates. Database Scanner should be used in conjunction with Internet Scanner and System Scanner. Internet Scanner and System Scanner can provide a detailed assessment and recommendations for locking down the base operating system and network services on which your database servers and related critical systems are hosted. Database Scanner is a single program that includes the functionality to scan Microsoft SQL Server, Oracle, and/or Sybase database servers.

##### Comments:

Database Scanner may provide a useful tool to verify the database requirements as established by the *Application Security Requirements Guide*. For example, Database Scanner assists in identifying the following types of vulnerabilities:

- **Authentication checks**-encompasses all the settings needed to verify each user’s claimed identity within the database management system. Includes password strength analysis, password aging, login attacks, stale logins, default login and password checks, and security of administrative accounts.
- **Authorization checks**-focuses on how an authenticated user is permitted to use specific resources within the system. Includes logon-hours violations, account and

## Draft

role permissions, stored procedure access, unauthorized object owners, resource access, and permissions.

- **System Integrity checks**-settings focused on coordinating and controlling system resources of the database system. Includes Trojan horses, OS integrity, audit configuration and analysis.

Database Scanner, like AppDetective, uses a simple GUI to automate the scanning and auditing process, and its results are presented in a clear manner.

### 4.2.3 OraScan

Product	<b>OraScan</b>
Vendor	Next Generation Software (NGSSoftware)
URL	<a href="http://www.nextgenss.com">www.nextgenss.com</a>

#### **Product Description:**

Please note that OraScan has not been publicly released (as of July 19, 2002). The following information is taken directly from NGSSoftware's promotional literature on its Web site.

“OraScan is part of NGSSoftware's intelligent Application Security Assessment Scanner suite of Next Generation security tools designed to completely automate the process of assessing an Oracle Web front end and its online applications. Although NGSSoftware originally intended to have OraScan ready for February 2002, [the company] took a major U-turn in the way [it] was designing the Oracle Web application assessment scanner. In the interim of starting OraScan and nearing its completion, [the company] decided to create a new technology, Pandora, that would reside at the core of our iASASs. Pandora is a proprietary technology that provides a hierarchical information store of scan checks, results and control elements. It allows NGSSoftware applications to interoperate by providing storage, persistence, and remoting facilities. The key benefits of Pandora are smaller, faster code, more flexible reporting, and a potential for distributed interoperation of NGSSoftware applications. Pandora promotes user extensibility, transparency, and easy upgrade paths with “Grafts.” On the technical side, Pandora will operate at the core of NGSSoftware's suite of intelligent application security assessment scanners, and its design provides for dynamic and diverse scanning based on previous results; Pandora components can base their actions on results provided by other components. Further, these components can be distributed across a network. {NCSSoftware} is, again, now nearing the completion of OraScan and aims to have a beta evaluation version available before May. This iASAS tool can audit bespoke PL/SQL, JSP,

## Draft

SQLJSP, and XSQL applications. Further, this OraScan examines the base server software for well-known and generic security issues.”<sup>2</sup>

### Comments:

None. As of July 19, 2000, the tool has yet to be publicly released.

---

<sup>2</sup> See <http://www.nextgenss.com/products/orascan.html>

## 4.3 DEVELOPER ASATS

### Overview:

Developer vulnerability assessment tools are designed to directly aid the application developer or software engineer. These tools are designed to locate potential vulnerabilities in either source code or compiled programs. They do not definitively find bugs; rather, they provide a reasonable starting point for performing manual security audits. These tools are composed of the following:

- **Source-Code Scanners**. These mainly open-source tools are very new and generally still under development. Their purpose is to scan source code, finding potentially dangerous function calls.
- **Fuzzers and Buffer Overflow Generators**. These tools test software by bombarding the program with random data. Note that most of these tools are open- source applications.

Note that these tools are intended for use by the developer or someone with extensive knowledge and understanding of the source code of applications. Although these tools are not inherently complex, they are sophisticated in that interpreting their results requires intimate knowledge of the code being analyzed. These tools will highlight potential vulnerabilities and allow the developer to either repair them or accept their risk.

The following tools are reviewed in this section:

- BFBTester
- CLint
- Cqual
- FlawFinder
- Fuzz
- ITS4
- Jlint
- PyChecker
- Rough Auditing Tool for Security (RATS)
- Splint

### Reviews:

#### 4.3.1 BFBTester

Product	<b>BFBTester</b>
Vendor	Open Source
URL	<a href="http://sourceforge.net/projects/bfbtester/">http://sourceforge.net/projects/bfbtester/</a>

### Product Description:

BFBTester is designed to produce efficient security checks of binary programs. BFBTester will analyze single and multiple argument command-line overflows and environment variable overflows. According to its developer, versions 2.0-beta and higher

**FOR INFORMATION PURPOSE**

can also watch for tempfile creation activity to alert the user of any programs using unsafe tempfile names. Although BFBTester is not intended to test all overflows in software, it is useful for detecting initial mistakes that can red flag dangerous software.

**Comments:**

BFBTester is a command-line driven program. Its results are also sent to standard output. As a result, a certain degree of expertise is needed to operate the program and interpret results. Moreover, this open-source program comes with little documentation-enough to outline its functionality and to get a user started. This tool is clearly meant for an expert-level user. BFBTester can be very Central processor unit (CPU) intensive as it creates multiple files in the process of trying to break an application. If BFBTester finds a potential flaw, the information reported, while succinctly conveying the result (for example, the tested program crashed when fed with a “-D” and a word 5,120 characters long), does not assist in developing a solution-it simply points out a potential flaw in the application where it may be susceptible to an overflow condition.

**4.3.2 CLint**

Product	<b>CLint</b>
Vendor	SourceForge (Open Source)
URL	<a href="http://sourceforge.net/projects/clint/">http://sourceforge.net/projects/clint/</a>

**Product Description:**

CLint is a testbed for static source-code checking techniques. It is designed to check C++ for common programmer errors and suggest improvements.

**Comments:**

CLint is an open-source tool clearly written for an expert-level user. Little documentation or usage information is provided with the release; therefore, it is largely up to the user to interpret the tool’s output.

**4.3.3 Cqual**

Product	<b>Cqual</b>
Vendor	University of California (Berkeley) (Open Source)
URL	<a href="http://www.cs.berkeley.edu/~jfoster/cqual/">www.cs.berkeley.edu/~jfoster/cqual/</a>

**Product Description:**

Cqual is a type-based analysis tool that provides a lightweight, practical mechanism for specifying and checking properties of C programs. Cqual extends the type system of C with extra user-defined type qualifiers. The programmer adds type qualifier annotations to his or her program in a few key places, and Cqual performs qualifier inference to check whether the annotations are correct. The analysis results are presented with a user interface that lets the programmer browse the inferred qualifiers and their flow paths. The technical idea behind Cqual is to perform constraint-based type inference. To analyze a program, Cqual traverses the program’s abstract syntax tree and generates a series of

## Draft

constraints that captures the relations between type qualifiers. A solution to the constraints gives a valid assignment of type qualifiers to the variables in the program. If the constraints have no solution, then a type qualifier inconsistency exists, indicating a potential bug.

### Comments:

Cqual is a well-developed and sophisticated tool for specifying and checking properties of C programs. This well supported tool comes with good documentation on its use. Cqual's parent project, the Open Source Quality Project at UC Berkeley, is geared to investigating techniques and tools for assuring software quality, such as finding and removing defects in software systems, and improving current methodology for designing high-quality software systems at the outset.

### 4.3.4 FlawFinder

Product	<b>FlawFinder</b>
Vendor	Open Source
URL	<a href="http://www.dwheeler.com/flawfinder/">www.dwheeler.com/flawfinder/</a>

### Product Description:

Developed by David Wheeler, Flawfinder is a python program that can be used to assist auditing C and C++ code. Flawfinder works by using a built-in database of C/C++ functions with well-known problems, such as—

- Buffer Overflow Risks (e.g., strepy, strcat, gets, sprintf, and the scanf family)
- Format String Problems (e.g., [v][f]printf, [v]snprintf, and syslog)
- Race Conditions (e.g., access, chown, chgrp, chmod, tmpfile, tmpnam, tempnam, and mktemp)
- Potential Shell Metacharacter Dangers (most of the exec family, system, and popen)
- Poor Random Number Acquisition (e.g., such as random)

Flawfinder produces a list of potential security flaws that are sorted by risk. This risk level depends not only on the function, but also on the values of the parameters of the function. For example, constant strings are often less risky than fully variable strings in many contexts. In some cases, Flawfinder may be able to determine that the construct is not risky at all, reducing false positives. Furthermore, Flawfinder correctly ignores text inside comments and strings (except for Flawfinder directives). Thus, Flawfinder gives better information-and better prioritization-than simply running “grep” on the source code.

### Comments:

Flawfinder is quite fast, covering thousands of lines of C code on a typical desktop machine within seconds. Flawfinder is released under GPL version 2 and therefore is free software. Note that not every potential coding error that Flawfinder detects is actually a security vulnerability, and not every security vulnerability is necessarily found. In fact, Flawfinder does not really “understand” the semantics of the code at all; it merely does

**FOR INFORMATION PURPOSE**

## Draft

simple text pattern matching. Nevertheless, Flawfinder can be a very useful aid in finding and removing security vulnerabilities.

### 4.3.5 Fuzz

Product	<b>Fuzz</b>
Vendor	SourceForge (Open Source)
Product Type	Developer Vulnerability Assessment Tool
URL	<a href="http://sourceforge.net/projects/fuzz/">http://sourceforge.net/projects/fuzz/</a>

#### **Product Description:**

Fuzz is a tool designed to attack certain kinds of software and find a particular type of programming error, where the programmer implicitly makes assumptions about the data stream that the program will be parsing. The concept is that if the data stream is substantially different from what the program expects, it might not be able to deal with it and might then crash. Although this approach has some limitations primarily because of the test data stream's random nature, the tool will test only a very small percentage of the total program. Fuzz will work on programs that receive and manipulate input from stdin (standard input).

#### **Comment:**

Fuzz is a tool meant for the expert developer and user. It runs from the command line and in order to interpret its results, requires a thorough knowledge of the application under development.

### 4.3.6 ITS4

Product	<b>ITS4</b>
Vendor	Cigital (Partial Open Source)
URL	<a href="http://www.cigital.com/its4/">http://www.cigital.com/its4/</a>

#### **Product Description:**

ITS4 (It's the Software, Stupid! [Security Scanner]) was developed to address the need for a practical, widely applicable tool to help people identify potentially unsafe constructs in C and C++ code. ITS4 performs only simplistic analysis on source code. ITS4 breaks a non-preprocessed file up into a series of lexical tokens, and then matches patterns in the stream of tokens. The user adds matching code; so non regular patterns can be recognized. ITS4 takes one or more C or C++ source files as input, breaking each into a stream of tokens. After scanning a file, ITS4 examines the resultant token stream comparing identifiers with a database of "suspects." ITS4 reads a vulnerability database from a text file at startup, keeping the entire contents resident in memory for the lifetime of the tool. Vulnerabilities can be added to the database, removed and changed with ease. The ITS4 vulnerability database contains more than 100 calls taken from many sources, including the Bugtraq archives, with the largest single class of problems being race conditions involving file accesses. Functions susceptible to buffer overflows also account for many entries. ITS4 was designed to allow easy manipulation and customization of its

**FOR INFORMATION PURPOSE**

## Draft

database and even allows programmers to specify functions to search for on a command line.

### Comments:

According to Cigital, the following are some initial conclusions about ITS4: ITS4 still requires a significant level of expert knowledge. Although ITS4 does encode a vast amount of knowledge on vulnerabilities that the developer no longer needs to keep in his head, it is still a tool that is better suited for an expert user who is able to take a potential vulnerability location and manually perform the static analysis necessary to determine if an exploit is possible. Moreover, even for experts, analysis is still time consuming. Because manual analysis is so time intensive, the tool only eliminates one-quarter of the time necessary for these analyses. However, ITS4 does assist in source code analysis and is capable of finding major problems. Cigital has identified several areas for improvement in the ITS4 design, such as trying to integrate the tool into various programming environments and trying to perform some sort of range analysis.

### 4.3.7 Jlint

Product	<b>Jlint</b>
Vendor	Open Source
URL	<a href="http://artho.com/jlint">http://artho.com/jlint</a>

### Product Description:

Jlint is designed to check Java code and locate bugs, inconsistencies, and synchronization problems. It actually consists of two separate programs performing syntax and semantic verification. The first of these, AntiC, is designed to fix problems with C grammar, which can cause dangerous programming bugs that go undetected by the compiler. AntiC is able to detect bugs such as the suspicious use of operator priorities, the absence of break-in switch code, and incorrect assumptions about construction bodies.

The second program, Jlint, is a semantic verifier and extracts information from Java class files. Jlint performs local and global data flow analyses, calculating possible values of local variables and catching redundant and suspicious calculations. By performing a global method invocation analysis, Jlint can detect the invocation of methods with possible “null” values. Jlint also builds a lock dependency graph for class dependencies and uses this graph to detect situations, which can cause deadlock during multi threaded program execution. In addition to deadlocks, Jlint can detect possible race condition problems, such as when different threads concurrently access the same variables.

### Comments:

Both AntiC and Jlint detect a wide variety of syntactic and semantic bugs in Java source code. Both programs are written in C++ and are designed to be system independent. The current release (v2.3) contains makefiles for UNIX systems with the gcc compiler and for Windows system with Microsoft’s Visual C++.

#### 4.3.8 PyChecker

Product	<b>PyChecker</b>
Vendor	SourceForge (Open Source)
URL	<a href="http://pychecker.sourceforge.net/">http://pychecker.sourceforge.net/</a>

##### **Product Description:**

PyChecker is a tool similar to Lint that discovers bugs in Python source code. It finds problems that are typically caught by a compiler for less dynamic languages, like C and C++. Because of the dynamic nature of python, some warnings may be incorrect; however, spurious warnings should be fairly infrequent.

PyChecker works in a combination of ways. First, it imports each module. The import provides some basic information about the module. The code for each function, class, and method is checked for possible problems.

The following types of problems can be found:

- No global found (e.g., using a module without importing it)
- Passing the wrong number of parameters to functions/methods/constructors
- Passing the wrong number of parameters to built-in functions and methods
- Using format strings that do not match arguments
- Using class methods and attributes that do not exist
- Changing signatures when overriding a method
- Redefining a function/class/method in the same scope
- Using a variable before setting it
- Self is not the first parameter defined for a method
- Unused globals and locals (module or variable)
- Unused function/method arguments (can ignore self)
- No doc strings in modules, classes, functions, and methods.

##### **Comments:**

PyChecker is a relatively new syntax analyzer for the Python programming language, reporting on constructions that can be compiled or browsed, respectively, but are still hazardous. This is important for Python because it, unlike more strongly typed languages such as C or Java, is highly susceptible to syntactic errors (yet it is this flexibility that makes the language appealing). PyChecker is able to detect and preclude potentially costly mistakes and has inspired much interest among Python programmers of all experience levels. PyChecker's validations include more sophisticated ones than just the spelling checks for uninitialized and unused variables. PyChecker is easy to set up and use, and its use offers a good return on investment.

#### 4.3.9 RATS

Product	<b>RATS</b>
Vendor	Secure Software (Open Source)

URL	<a href="http://www.securesw.com/rats.php">http://www.securesw.com/rats.php</a>
-----	---

**Product Description:**

RATS is developed by Secure Software Solutions and is a security auditing utility for C, C++, Python, Perl, and PHP code. RATS scans source code, finding potentially dangerous function calls such as buffer overflows and Time Of Check, Time Of Use (TOCTOU) race conditions. The goal of the RATS project is not to definitively find bugs, but to provide a reasonable starting point for performing manual security audits. As its name implies, the tool performs only a rough analysis of source code. It will not find every error and will also find items that are not errors.

**Comments:**

Although RATS is released under version 2 of the GNU Public License (GPL), it is clearly meant for the expert user because as little documentation is provided. The tool is fairly flexible, however, in terms of defining the type of vulnerabilities it uncovers. The user can specify what vulnerabilities are reported in the tool's output report via both the data contained in the vulnerability databases that are used and the specified setting of the tool's vulnerability warning level. For each vulnerability, the list of files and line numbers where it occurred is provided, followed by a brief description of the vulnerability and suggested action.

**4.3.10 Splint**

Product	<b>Splint</b>
Vendor	University of Virginia (Open Source)
URL	<a href="http://lclint.cs.virginia.edu/">http://lclint.cs.virginia.edu/</a>

**Product Description:**

Splint is a tool for statically checking C (but not C++) programs for security vulnerabilities and coding mistakes. Splint does many of the traditional Lint checks, including unused declarations, type inconsistencies, use before definition, unreachable code, ignored return values, execution paths with no return, likely infinite loops, and fall-through cases. More powerful checks are made possible by additional information given in source code annotations. (Annotations are stylized comments that document assumptions about functions, variables, parameters, and types.) In addition to the checks specifically enabled by annotations, many of the traditional Lint checks are improved by exploiting this additional information. (Note that before 2002, Splint was known as LCLint. Splint 3.0 is the successor to LCLint 2.5.)

The following types of problems are detected by Splint:

- Dereferencing a possible null pointer
- Using possibly undefined storage or returning storage that is not properly defined
- Type mismatches, with greater precision and flexibility than provided by C compilers
- Violations of information hiding

## Draft

- Memory management errors, including uses of dangling references and memory leaks
- Dangerous aliasing
- Modifications and global variable uses that are inconsistent with specified interfaces
- Problematic control flow such as likely infinite loops, fall through cases or incomplete switches, and suspicious statements
- Buffer overflow vulnerabilities
- Dangerous macro implementations or invocations
- Violations of customized naming conventions.

### Comments:

Splint is a highly developed and sophisticated tool for finding vulnerabilities in C code. The tool, which is well supported by the Secure Programming Group at UVA, comes with extensive documentation on its use. It was written in C; therefore, anyone with a standard American National Standards Institute (ANSI) C compiler can recompile it for their platform. Splint is a versatile tool because it can analyze code at many levels of abstraction. Splint has several levels of analysis, corresponding to a greater or smaller quantity of different checking techniques.

## 4.4 GENERAL-PURPOSE ASAT'S

### Overview:

General-purpose vulnerability assessment tools scan networks and systems for potential security weaknesses and recommend fixes. Because these tools are designed to scan a broad range of software and hardware, their ability to focus in depth on the vulnerabilities of a specific item, such as a database, is limited.

The following tools are reviewed in this section:

- Bv-Control
- FoundScan
- Nessus
- NetRecon
- Retina
- SAINT
- Security Analyzer
- STAT Scanner
- Typhon II.

### Reviews:

#### 4.4.1 Bv-Control

Product	<b>Bv-Control</b>
Vendor	Bindview
URL	<a href="http://www.bindview.com/">http://www.bindview.com/</a>

### Product Description:

Bv-Control is a tool for scanning Windows 95/98/NT/2000, UNIX, and network devices from a host platform of either Windows NT or 2000. The tool scans for known vulnerabilities, including the SANS top 10 vulnerability list, as well as known vulnerable CGI programs. Bv-Control works by creating a map of all servers, workstations, and IP devices in the testing space. Specific IPs may also be entered in the case that they are not automatically found by the tool. The tool will then probe each device, testing for security vulnerabilities that are contained within Bv-Control's database. Bv-Control can also scan the OS and internal configuration of all Windows NT and 2000 servers. This includes checking for missing operating system patches, integrity of key system files, file directory permissions, and registry values and permissions.

During the scanning progress, a real-time progress indicator shows a summary of the security holes found and the current progress. In addition, alerting capabilities help notify users when certain security holes are found.

Bv-Control takes all of the findings and, using a Crystal Reports print engine, generates reports for the executive, management, and administrator audiences. In addition, there are features to help in the filtering of specific security holes in the reports or the

**FOR INFORMATION PURPOSE**

## Draft

comparison between a current scan and an earlier baseline scan against the target. When multiple scans and reports are run over time, there is also the capability to consolidate the data from multiple reports to allow a better overall view of the environment scanned.

### Comments:

Although the scanner documentation makes much mention of being certified by the SANS Institute to check all aspects of the SANS Top 10 Vulnerabilities List, it must be noted that the SANS Top 10 Vulnerability List was changed to form the top 20 list. It is unknown if this change will be reflected in the Bv-Control product at the time of this writing. Bv-Control allows for the easy creation of an overall network security posture for known vulnerabilities, especially as time progresses and additional scans are performed. Not only does the tool allow the comparison of scans over time, but it can also consolidate multiple sub scans for the network and scans of additions to the overall network. Along with the many levels of reporting and the use of crystal reporting, accurate and informative reports can be created that help to correct any security issues. The easy configuration and scanning of networks, along with the extra functionality in Windows NT and 2000 servers, make this a good scanner for known vulnerabilities in Windows environments.

### 4.4.2 FoundScan

Product	<b>FoundScan</b>
Vendor	Foundstone Inc.
URL	<a href="http://www.foundstone.com">http://www.foundstone.com</a>

### Product Description:

FoundScan is a network security scanner that not only scans for vulnerabilities, but also creates a detailed view of the network that is scanned, including details of the hosts, network services, OSs, and a network map. In addition to all these details, the scanner can run automatically on a scheduled basis or continuously so that short- and long-term trending can be determined. This is made even easier with the graphical reports that are generated, showing the changes in what is termed the FoundScore.

The FoundScore is a score 0 to 100 that FoundScan assigns as an overall risk level based on the information gathered and known vulnerabilities discovered in the scanned targets. Each aspect of the assigned score included detailed explanations of how the score was reached in addition to detailed instructions explaining how to fix a network so that it scores better.

After mapping the network and discovering all the services, OSs, details of a network and its components, FoundScan uses intelligent testing techniques to ensure a quick and accurate analysis of the vulnerabilities present. This intelligent scanning includes using information found in each vulnerability to help exploit other vulnerabilities, chaining together all discovered information. FoundScan also uses a unique methodology to completely assess custom e-commerce applications, finding design flaws and security risks for which normal general vulnerability scanners do not test.

**FOR INFORMATION PURPOSE**

## Draft

In addition to the assignment of the FoundScore, a complete report with information and remediation strategy for all the vulnerabilities is created. Reporting capabilities are also configurable for e-mail alerting. All vulnerabilities tested for are updated regularly and when found during a test are tracked from identification through a resolution, with all resolution information provided.

### Comments:

FoundScan is an intelligent scanner that is extremely quick while remaining accurate. Note that FoundStone—an important and active participant in the information security arena, designs this tool. They are most well known for their *Hacking Exposed* series of books in recent years and active participation in the hacking community.

The reports generated by this tool are highly graphical, allowing for quick summaries of the overall security status of networks while also including all the details and functionality to track vulnerabilities from the discovery until the remediation of the vulnerability. FoundScan is an easy-to-use, yet powerful tool for mapping a network's security posture while simultaneously discovering and fixing the known vulnerabilities. The tool contains limited features for discovering unknown vulnerabilities, such as SQL query poisoning, exposed sensitive files, improper input validation, source code disclosure, easily guessed passwords, and other e-commerce application vulnerabilities.

### 4.4.3 Nessus

Product	Nessus
Vendor	Nessus
URL	<a href="http://www.nessus.org/">http://www.nessus.org/</a>

### Product Description:

Nessus is a free, open-source vulnerability scanner that is maintained and updated constantly by a community of developers. It is an extremely powerful scanner with a plug-in architecture allowing all security checks to be written in Nessus Attack Scripting Language (NASL). NASL is a language designed for Nessus to allow additional security checks to be written quickly and without knowledge of the Nessus engine. Security checks for known vulnerabilities are updated daily and are freely available for download at any time.

The Nessus scanner is made of two parts: a server that performs the scanning and attacks, and a client that serves as the front end. The client and server can operate on different machines, and multiple clients may be used. Nessus uses a smart service recognition strategy to ensure that services on non standard ports are still identified and tested for the appropriate vulnerabilities. In addition, options are available to ensure only the correct vulnerability checks are run against correspondingly vulnerable applications. In addition, other means exist for controlling whether dangerous or potentially dangerous vulnerability checks are employed. These means range from a broad setting that

## Draft

eliminates all dangerous checks, down to a more granular setting that allows the user to select each individual test that will be run.

The Nessus scanner can scan an unlimited number of hosts (depending on server hosting hardware) in a quick and parallel method. Nessus can then create a complete and informative report documenting the vulnerabilities found in addition to the remediation strategies for the particular vulnerabilities.

### Comments:

The Nessus scanner is a community-supported scanner that is constantly updated based on the security needs of its user base. If updates are not soon enough, the NASL plug-ins allow for full customization of the tests without knowledge of the specifics of the Nessus engine. Although these features exist to allow for a powerful and customizable tool, the level of expertise required for full use of the scanner is not that of a point-and-click environment. The installation of the scanner and customization of the tests to be performed requires a high level of knowledge. However, once this effort is accomplished, the tool can function with the same ease as most scanners that check for known vulnerabilities. The reports are complete and contain enough clear instructions for the correction and fixing of the vulnerabilities. Some expert analysis may be needed to determine which vulnerabilities are false positives, however.

#### 4.4.4 NetRecon

Product	<b>NetRecon</b>
Vendor	Symantec
URL	<a href="http://enterprisesecurity.symantec.com/products">http://enterprisesecurity.symantec.com/products</a>

### Product Description:

NetRecon is a vulnerability assessment scanning tool that uses a progressive scanning technique. This progressive scanning technique allows NetRecon to gather vulnerability information in parallel and share the information between components. This allows vulnerability findings on one computer to be used adaptively to further test other computers in the testing range.

NetRecon is capable of scanning UNIX, Linux, Windows 95/98/NT/2000, and NetWare machines. It is also able to use such protocols as TCP/IP, Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX), and NetBEUI and to test devices such as firewalls, routers, and other network devices.

Not only does NetRecon identify vulnerabilities found during scanning, but it also details the path taken to uncover the vulnerability, known as “root cause and path analysis” by NetRecon. NetRecon then identifies the correct place to fix the vulnerabilities that have been found and provide the detailed fix information.

NetRecon can produce graphical reports along with either technical or executive reports that can further be customized using Crystal Reports. All data found during the scan

**FOR INFORMATION PURPOSE**

## Draft

process is displayed real-time, so there is no need for the scanner to wait for completion to start reviewing the findings.

The LiveUpdate feature within NetRecon allows for an easy update to the most current vulnerability releases from Symantec before any scans are performed. There is no set frequency of when updates are available, but history of the past year shows 10 updates for NetRecon 3.5 from July 23, 2001, to July 2, 2002, ranging from about 2 weeks between updates to as infrequent as almost 3 months.

### Comments:

With its use of progressive scanning, NetRecon takes on the role of an attacker, gathering information for each individual scanned device, but then trading the information to see if further levels of compromise can be obtained. This action is clearly detailed and explained in the tool “root cause and path analysis” section of the report, which helps provide the user a clear understanding of how the security or insecurity of each device can affect some or all of the other devices. This is an often-overlooked aspect of security when evaluating the current security posture of an environment.

### 4.4.5 Retina

Product	Retina
Vendor	Eeye
URL	<a href="http://www.eeye.com">http://www.eeye.com</a>

### Product Description:

Retina is designed to scan for both known and unknown vulnerabilities quickly, concisely, and accurately. Retina accomplishes this using an artificial intelligence (AI) module, which analyzes all data gathered to determine exactly what protocol and service is running before proceeding and running the appropriate vulnerability checks.

To find possible unknown vulnerabilities, Retina uses a feature labeled Common Hacking Attack Methods (CHAM), along with the AI technology, to attempt to identify potentially unknown vulnerabilities. In particular, the CHAM feature will test for buffer overflows in software and applications that may allow for exploitation with further testing and coding.

Retina includes vulnerability auditing modules for the following systems and services:

- NetBIOS
- HTTP
- CGI
- WinCGI
- File Transfer Protocol (FTP)
- Domain Name Server (DNS)
- DOS
- TCP/IP and User Datagram Protocol (UDP)
- Registry
- Services
- Users and Accounts
- Password Vulnerabilities
- Publishing Extensions
- Database Servers

**FOR INFORMATION PURPOSE**

## Draft

- Post Office Protocol (POP)
- Simple Mail Transfer Protocol (SMTP)
- Lightweight Directory Access Protocol (LDAP)
- Firewalls
- Routers
- Proxy Servers.

In addition to including detailed fix information for all of the above known systems and service vulnerabilities, Retina can automatically fix many of the above listed, such as registry settings and file permissions.

As new vulnerabilities are discovered, an auto-update feature provides regular updates for its modules. For individuals who discover their own vulnerabilities or desire other modules to be checked, Retina also has an open architecture that allows users to create their own modules with any programming language, such as Perl, C, C++, Visual Basic, and Delphi.

### Comments:

Retina's use of the AI to perform smart scanning allows for an extremely fast scanning tool. This smart scanning feature also helps prevent potentially dangerous vulnerability testing for services that are not actually present. In addition to the AI functionality, the CHAM feature is very helpful in discovering unknown buffer overflows in applications. (Note that this is the same CHAM feature that has allowed eEye to discover some of the more recent large vulnerabilities in Microsoft IIS.) These features are all easily and intuitively accessed through a well-designed GUI. All aspects of the scanning and reporting are easily controlled through this GUI also. Although the reports generated by Retina are very powerful in content, they are limited to only a technical or executive report in HTML format.

### 4.4.6 SAINT

Product	<b>SAINT</b>
Vendor	Saint Corporation
URL	<a href="http://www.saintcorporation.com/">http://www.saintcorporation.com/</a>

### Product Description:

SAINT scanner is an open-source vulnerability based off of Security Administrator Tool for Analyzing Networks (SATAN), an early, open-source vulnerability assessment tool. Current versions of SAINT have been commercialized, however. The SAINT Corporation allows for the free download of the scanner, but only with the purchase of either SAINTreporter or SAINTwriter. These tools add advanced reporting features and automatic update capabilities, respectively. (Note that these extra features are not addressed in the scope of this document.)

SAINT is a UNIX based tool that works in most UNIX environments to scan all types of remote hosts and networks for vulnerabilities. The tool contains a Web-based GUI; however, not all configuration methods are available in this GUI. Configuration is also

**FOR INFORMATION PURPOSE**

## Draft

necessary by the manual editing of a file, `saint.cf`, which is not always an intuitive process.

All scanning is performed manually, unless the product is run from the command line along with the UNIXZ utility *cron*. Custom probes can be designed and implemented within the SAINT architecture, so the scan types are limitless. This also allows for the manipulation or editing of the current vulnerability checks. The vulnerabilities scanned for can include the SANS Top 20, and all include detailed background information and possible fix information when available. Vulnerabilities scanned for are linked to Common Vulnerabilities and Exposure (CVE) information when scanned for.

### Comments:

The vulnerabilities for which SAINT scans are usually limited to those vulnerabilities with available fixes. SAINT will not identify all potential risk areas of an environment. However, with those vulnerabilities identified, the tool will then provide detailed information about the vulnerability in addition to appropriate fix or remediation guidance.

Although the Web interface for use of the tool appears simple and straightforward, this can be deceptive. Much of the tool's configuration is changed via editing a configuration file, which could lead to confusion. Improper configuration can allow for easy mistakes, such as scanning remote targets outside the approved area. Including targets that are not owned or controlled by the user. This action is possible as a result of scanning methodology used by SAINT—a multi pass scan that uses information gathered from previous scans to further find vulnerabilities and to discover all trust relationships between computers. These trust relationships will sometimes cross the boundaries of the desired scanning range unless explicit configurations are made in `saint.cf`.

Some configuration capabilities are now included in the GUI Web interface; however, careful thought should be placed in the configuration selections to ensure the scanner will perform as desired before any actual scanning is started. Overall SAINT can be a useful vulnerability scanner in enhancing network security, but it should not be used if expecting all known vulnerabilities to be identified. In addition, good documentation is available to guide with the possible configuration difficulties, which should improve the learning curve needed to use the tool.

### 4.4.7 Security Analyzer

Product	Security Analyzer
Vendor	NetIQ
URL	<a href="http://www.netiq.com/products/sa/">http://www.netiq.com/products/sa/</a>

#### Product Description:

Security Analyzer is a vulnerability assessment product for Windows, Solaris, and Linux platforms. In addition to identifying vulnerabilities and providing detailed correction information, Security Analyzer provides comparative reports so that tracking of security improvements is made easy and apparent over time.

**FOR INFORMATION PURPOSE**

## Draft

In addition to performing network-based scans, security agents can be deployed to perform host-based scans. Host-based scans allow for full and comprehensive registry, service, and application data gathering from the agent computers, along with distributed processing for quick and efficient network wide scans.

Security Analyzer tests for a wide range of vulnerabilities and includes many different scan profiles with multiple testing policies. Each policy can be edited to include or remove general categories of testing down to individual vulnerability checks. In addition, there is a Software Developers Kit (SDK) to enable the expert user to create custom tests for unique environments.

The reporting features are accessible immediately within the tool, or by creating customizable HTML or Word reports. Using previous scans and reports, comparative reports can be generated to show improvements or new risks in the environment after changes are made.

### Comments:

The Security Analyzer is a powerful GUI tool that is highly configurable in all aspects of testing and reporting. The many features of the tool are intuitive to use in the GUI, but the sheer number of testing configurations that can be established require a longer learning process and scan set up time compared with other tools. This is a very powerful tool for use in not only maintaining a secure posture for the specific OSs for which the tool scans for but also comparing the vulnerability findings with the security policy of an environment. Custom testing policies can be created to match a security policy and test for compliance with ease.

### 4.4.8 STAT Scanner

Product	<b>STAT Scanner</b>
Vendor	Harris
URL	<a href="http://www.statonline.com/">http://www.statonline.com/</a>

### Product Description:

STAT Scanner is a general vulnerability scanner for Windows NT/2000/XP and UNIX/Linux environments. STAT Scanner can scan individual computers or networks while automatically fixing a large portion of vulnerabilities that it detects with an AutoFix feature for the Windows NT or 2000 computers.

The reporting features allow for the presentation of detailed information on the vulnerabilities found along with remediation strategy and an automatic fix capability when available. The tools allow for current and previous scans to be analyzed for trends and will generate management- or technical-style reports. Generated reports also include graphical summaries of the overall security posture based on the information gathered

## Draft

and vulnerabilities found Updates for newly discovered vulnerabilities are available monthly.

### Comments:

STAT Scanner is a simple scanner that is limited in the number of computers or devices it can scan resulting from the OS specific requirements of the targets it will scan. Although the tool allows for detailed customization of each individual vulnerability that is scanned for, the process of editing the profiles can be time consuming, making it difficult to understand what general vulnerability classes will be scanned for in the final configuration profile. Many different levels of reporting are available, from high-level summaries to the detailed vulnerability reports, along with the capability to compare reports of different scans, but no further customization is available. These reports do appear to be very detailed though, with many references to outside information such as CVE, Bugtraq, SANS, and other information and advisories. Unfortunately, the update of vulnerabilities scanned for are available only monthly. This may allow for an outdated or inaccurate vulnerability assessment report.

### 4.4.9 Typhon II

Product	<b>Typhon II</b>
Vendor	Next Generation software
URL	<a href="http://www.nextgenss.com/">http://www.nextgenss.com/</a>

### Product Description:

Typhon II is the commercial version of Typhon I, which was an update of the free scanner Cerberus Internet Scanner. Typhon II is a vulnerability scanner that runs from a Windows NT or 2000 host, scanning for host-based vulnerabilities in addition to containing a war dialing functionality.

Typhon II scans a host or range of hosts to identify what services are offered. It then probes each service found to determine which services contain vulnerabilities. With Windows NT and 2000 machines, this effort includes a detailed registry module check among other specific Windows modules.

Typhon II checks include tests for the following:

- Web checks
- SMTP checks
- FTP checks
- POP3 checks
- Simple Network Management Protocol (SNMP) checks
- Remote Procedure Call (RPC) checks
- DNS checks
- NT service checks
- SQL service checks
- IE browser checks
- NT audit checks
- Protocol checks
- RServices checks
- LDAP checks

**FOR INFORMATION PURPOSE**

## Draft

- Finger checks
- NetBIOS checks
- NT registry checks
- Oracle checks
- Secure Shell (SSH) checks.

Typhon II creates a simple HTML file, which is indexed by the primary scan categories performed on the system. Each section contains detailed information on the findings, including the effect of vulnerabilities and any fix information.

### Comments:

Typhon II is a simple-to-use, GUI-based tool that checks for many standard vulnerabilities in addition to enumerating much Windows information. This useful and quick auditing tool is perhaps more appropriately used as a supplemental tool when performing vulnerability checks. Generalized configuration changes are easily and intuitively made with the provided interface. It is not a complete vulnerability checker, however, and should not be used stand-alone if a complete inventory of known vulnerabilities is desired. The reporting feature documents each host individually; therefore, when scanning a large number of hosts, a review of the findings can be time-consuming and tedious, especially if attempting to correct the entire findings.

## 4.5 OTHER USEFUL TOOLS

### Overview:

This is a catchall category to capture interesting and potentially useful tools that may aid in providing application and database security. They comprise a range of tools, which are not directly used for vulnerability assessments, but may either assist in securing a site or complement the vulnerability assessment process. (Note that additional commentary is provided when appropriate.)

The following tools are reviewed in this section:

- Achilles
- AppShield
- Blast
- IDA Pro Disassembler
- Metis
- PatchLink Update
- SecurityExpressions
- STAT Analyzer
- STAT Neutralizer
- StormWatch
- StormFront
- Web Proxy.

### Reviews:

#### 4.5.1 Achilles

Product	Achilles
Vendor	DigiZen
Product Type	Web Proxy Tool
URL	<a href="http://www.digizen-security.com">www.digizen-security.com</a>

### Product Description:

Achilles is a Microsoft Windows platform Web application security tool that serves as a proxy server, acting as the man-in-the-middle during an HTTP session. Achilles will intercept an HTTP session's data in either direction and enable the user to alter the data before transmission. For example, during a normal HTTP SSL connection, a typical proxy will relay the session between the server and the client and allow the two end nodes to negotiate SSL. In contrast, when in intercept mode, Achilles will pretend to be the server and negotiate two SSL sessions; one with the client browser and the other with the Web server. As data is transmitted between the two nodes, Achilles decrypts the data and enables the user to alter and/or log the data in clear text before transmission.

### Comment:

**FOR INFORMATION PURPOSE**

## Draft

Achilles, like WebProxy, is intended for the expert user (as it serves as a proxy server and thus all malformed data must come from the user) and may assist in the discovery of unknown vulnerabilities within a Web application by providing a simple means of intercepting and modifying communication between the client and server.

### 4.5.2 AppShield

Product	<b>AppShield</b>
Vendor	Sanctum Inc.
Product Type	Web Application Firewall
URL	<a href="http://www.sanctuminc.com">www.sanctuminc.com</a>

#### Product Description:

AppShield is an active system that monitors and responds to any unusual or unauthorized behavior within a site, theoretically blocking attacks before they can reach the site. AppShield software is a Web application firewall built on a security proxy architecture that enforces a “positive security model” blocking any type of application manipulation. (The positive security model enforces intended behavior versus watching for unintended behavior.) AppShield works by creating, automatically and on the fly, rules for legitimate behavior based on the HTML code within the page. It can then check that every request conforms to the specific policy for that user session and page. The idea is that Web applications (both the application and Web logic) can be used only the way they were intended by the developer. Any attempt at manipulating them is then directly blocked.

#### Comment:

Although not directly a vulnerability assessment tool, AppShield will assist in securing Web sites and applications by limiting behavior. Because many unknown vulnerabilities within Web applications stem from unintended responses to irregular input, AppShield will provide an added layer of defense by filtering input before it reaches an application.

### 4.5.3 Blast

Product	<b>Blast</b>
Vendor	FoundStone (Open Source)
Product Type	Developer Vulnerability Assessment Tool
URL	<a href="http://www.foundstone.com/knowledge/proddesc/blast.html">www.foundstone.com/knowledge/proddesc/blast.html</a>

#### Product Description:

Blast is a small TCP service stress test tool. Blast can help spot potential weaknesses in network servers. Essentially, Blast works by sending a large user-specified character string to a particular IP address and port. Blast comes with little documentation, and it is largely up to the user to determine how to use the tool and interpret its results.

#### Comment:

Very little documentation is included with this tool, which is clearly meant for the expert user.

**FOR INFORMATION PURPOSE**

#### 4.5.4 IDA Pro Disassembler

Product	<b>IDA Pro Disassembler</b>
Vendor	DataRescue
Product Type	Developer Vulnerability Assessment Tool
URL	<a href="http://www.datarescue.com/idabase/">www.datarescue.com/idabase/</a>

##### **Product Description:**

IDA is an interactive disassembler that allows the user to actively participate in the disassembly process. (Note that IDA is not an automatic analyzer of programs, but it does assist in identifying suspicious instructions and unsolved problems.) IDA Pro is designed to revert a compiled program back into intelligible source code. IDA Pro disassembles programs compiled for more than 30 microprocessors, automatically distinguishing code from data. The tool's Fast Library Identification and Recognition Technology (FLIRT) module quickly identifies library functions from common compilers. When the user uncovers facts, the application propagates them back through the source code, thereby speeding the reverse engineering process.

##### **Comment:**

This tool, although somewhat questionable ethically, will potentially allow older "legacy" applications, which lack original source code, to be reverse engineered. In doing so, potential vulnerabilities within the "original" source code may be uncovered and subsequently mitigated.

#### 4.5.5 Metis

Product	<b>Metis</b>
Vendor	.severus.org (open source)
Product Type	Web Site Data Collection Tool
URL	<a href="http://www.severus.org/sacha/metis/">www.severus.org/sacha/metis/</a>

##### **Product Description:**

Metis is a tool to collect information from the content of Web sites. It was written for the Ideahamster Group for finding the competitive intelligence weight of a Web server and assists in satisfying the CI Scouting portion of the Open Source Security Testing Methodology Manual (OSSTMM)(see <http://www.ideahamster.org/osstmm-description.htm>). The tool is written in Java and distributed under the GNU Public license. The tool is composed of two packages: the Web spider engine, which handles the Web spidering process, collecting and storing information in memory; and the data analysis package which reads the data collected by the spider and generates a report.

#### 4.5.6 PatchLink Update

Product	<b>PatchLink Update</b>
Vendor	PatchLink Corporation

**Draft**

Product Type	Automated Patching Tool
URL	<a href="http://www.patchlink.com">www.patchlink.com</a>

**Product Description:**

PatchLink Update is cross-platform patch discovery and distribution utility that provides enterprise wide patch, software, data, and task deployment across the Internet. It is designed to automatically detect patch-related security vulnerabilities on all machines within a network and provide a simple means to immediately correct them across all platforms and enterprise boundaries. PatchLink performs an enterprise wide discovery of software and patch configurations on all machines within the network, subsequently reporting the version and date of existing patches and any missing patch on each computer. The system administrator can then disregard the unwanted patches and remove them from future reporting or choose to download and deploy the critical patches to the entire network or only a selected group of computers.

**Comment:**

Many vulnerabilities stem from outdated and unpatched versions of software running on servers. PatchLink will provide an efficient tool for ensuring that all software within an enterprise is up-to-date or at least informs system administrators where potential vulnerabilities may exist.

**4.5.7 SecurityExpressions**

Product	<b>SecurityExpressions</b>
Vendor	Pedestal Software
Product Type	Policy Compliance Testing and Fixing Tool
URL	<a href="http://www.pedestalsoftware.com">www.pedestalsoftware.com</a>

**Product Description:**

Security policies, although typically comprising a set of industry best practices, must be tailored to specific IT environments. SecurityExpressions ensures that systems comply with organizational security policies. As an auditing tool, it checks permissions, accounts, settings, patches, and other vital security options without requiring any agent software on Windows, Solaris, Linux, and other UNIX platforms. It allows system and security administrators to define a set of organizational policies based on industry best practices and local requirements and to ensure consistent implementation of those policies across the enterprise. Policies are implemented as a set of rules in Security Information File (SIF) files. SecurityExpressions is distributed with a default set of SIF files, including IE, Microsoft Fixes, Microsoft Security White Paper for NT, NSA Guidelines for Windows 2000, SANS Securing Windows NT Step-by-Step Guide, and the U.S. Navy Guide for Securing Windows NT. For example, the IE SIF contains rules specific to the IE browser configuration, whereas the Microsoft Fixes SIF has a set of rules to determine the status of hotfix installation. The tool provides instant remedies through its reporting capabilities, allowing administrators to click on any line in a basic report to amend rules for bringing particular systems and groups into compliance. Alternatively, the application can automatically change noncompliant parameters to the values specified in the ruleset.

**FOR INFORMATION PURPOSE**

History logs of all actions are maintained so that any changes can be reversed, if necessary, to allow comparative reporting over time.

#### 4.5.8 STAT Analyzer

Product	<b>STAT Analyzer</b>
Vendor	Harris Corporation
Product Type	Automated Risk Assessment Framework
URL	<a href="http://www.statonline.com">www.statonline.com</a>

##### **Product Description:**

STAT Analyzer is designed to automate the security assessment process, using a workflow typical employed among security engineers. The software first obtains a model of a network using either the integrated network discovery or modeling tool (STAT Scanner) or third-party tools. STAT Analyzer allows the user to tailor a security policy against which the network can be assessed. Analyzer then runs the vulnerability scanners and analysis tools, consolidating and merging the vulnerability assessment and analysis tool results and analyzing the network according to the specified security policies. STAT Analyzer works with other network modeling (e.g., Nmap, WhatsUp Gold) and vulnerability assessment tools (e.g., Nessus, ISS Internet Scanner) to perform a complete network security assessment. It automates these tools to work together and complement each other. It then uses a fuzzy logic based process to further correlate and analyze these tool outputs to improve results.

#### 4.5.9 STAT Neutralizer

Product	<b>STAT Neutralizer</b>
Vendor	Harris Corporation
Product Type	Intrusion Prevention Tool
URL	<a href="http://www.statonline.com">www.statonline.com</a>

##### **Product Description:**

STAT Neutralizer is designed to provide intrusion prevention from damage caused by malicious code, internal or external attacks, and human error. The tool works by intercepting bad behavior at the lowest possible level of the OS, the kernel. Based on a set of rules administered through a centralized administrative interface, the intercepted action can be allowed, denied, terminated, and/or logged. STAT Neutralizer watches for bad behavior occurring at the host level and is not subject to circumvention by encryption, fragmentation, or other common network attacks. STAT Neutralizer offers a standard set of security policies that are “customizable” with an optional toolkit, allowing them to be customized to specific environment and organizational needs. The intrusion response can also be tailored to allow the administrator to determine how much or how little prevention is desired, securing applications and files that would not normally be protected. STAT Neutralizer supports Windows NT® 4.0 and Windows® 2000. Support for Windows® XP is forthcoming.

**Comment:**

Similar to Okena’s StormWatch or Sanctum’s AppShield, STAT Neutralizer will assist in securing Web sites and applications by limiting behavior. STAT Neutralizer will provide an added layer of defense by filtering potentially dangerous behavior and input before it reaches an application.

**4.5.10 StormWatch and StormFront**

Product	<b>StormFront and StormWatch</b>
Vendor	Okena
Product Type	Intrusion Prevention Tools
URL	<a href="http://www.okena.com">www.okena.com</a>

**Product Description:**

StormFront works with StormWatch by analyzing any business application’s actual operation and automatically creating a StormWatch protective policy to secure it. Because this analysis is based on actual, observed behavior, it allows protection of any application. StormFront automatically configures StormWatch agents to monitor an application’s behavior and collects the information needed to build a policy.

StormWatch is designed to be an intrusion prevention tool that defends a network by deploying intelligent agents across desktops and servers to ensure their integrity. Unlike an intrusion detection tool, StormWatch agents intercept an application’s resource requests to the OS so it can make real-time allow/deny decisions according to the application security policy set by the system owner. StormWatch takes a different approach by defining “good” behaviors and then enforcing those behaviors on every end-user desktop and network server across an enterprise. Administrators configure rules that control which actions applications can perform on file, network, and system resources for each system on which StormWatch is installed. System calls are then intercepted and correlated against defined policies and rules. (Note: although several default policies are included with the product, a user can define and develop new rules according to business practices.) As a result of this correlation, the operation is either allowed or denied according to security policy.

**Comment:**

Similar to Harris’ STAT Neutralizer or Sanctum’s AppShield, Okena’s products will assist in securing Web sites and applications by limiting behavior. Because many unknown vulnerabilities within Web applications stem from unintended responses to irregular input, StormFront will “learn” acceptable behavior and allow StormWatch to provide an added layer of defense by filtering input before it reaches an application.

**4.5.11 WebProxy**

Product	<b>Web Proxy</b>
Vendor	@Stake
Product Type	Web Proxy Tool

## Draft

URL	<a href="http://www.@stake.com/research/tools">www.@stake.com/research/tools</a>
-----	--

### **Product Description:**

WebProxy is a cross-platform/browser security tool for use in auditing Web sites. WebProxy is installed as a proxy for a Web browser, allowing the user to intercept, modify, log, and resubmit requests (HTTP and HTTPS). Editing capabilities include the parsing of query parameters, request headers, and POST parameters, as well as cookie editing. WebProxy allows on-the-fly editing of HTTP requests based on regular expression matching. WebProxy will also dynamically generate certificates. WebProxy can be used for SQL injection, cookie manipulation, parameter testing, or simply monitoring requests.

### **Comment:**

WebProxy, like Achilles, is intended for the expert user (because it serves as a proxy server, all malformed data must come from the user) and may assist in the discovery of unknown vulnerabilities within a Web application by providing a simple means of intercepting and modifying communication between the client and server.

## **5. CONCLUSION**

Application security assessment tools provide vital information to developers and technical staff, ranging from the identification of known vulnerabilities, to the discovery of potential flaws, and to the recommendation of mitigation strategies and repair techniques. These tools serve an important role in maintaining the security of an application by providing some assurance that the application complies with and, when necessary, enforces, all security policies governing the application itself, the data it handles, the system to which it belongs, its operating environment, and its users.

Based on this market survey, it is clear that no single tool can single-handedly provide a complete assessment solution for assuring the security of an application. Most ASATs will perform admirably in assessing vulnerabilities; however, none of the tools completely address the application security requirements put forth in the *Recommended Standard Application Security Requirements* document that was issued as the first in this application security series. Therefore, as the overall task of building an Application Security Compliance Toolkit moves forward, it will be important to scan the varied categories of ASATs and select several of the most appropriate tools for inclusion. Fashioning the toolkit in such a manner will allow the maximum number of security requirements to be addressed and will provide the best possible solution.

## 6. REFERENCES

### Web Application ASATs

AppScan	<a href="http://www.sanctuminc.com">www.sanctuminc.com</a>
DominoScan	<a href="http://www.nextgenss.com">www.nextgenss.com</a>
HailStorm	<a href="http://www.cenzic.com">www.cenzic.com</a>
N-Stealth	<a href="http://www.nstalker.com">www.nstalker.com</a>
Nikto	<a href="http://www.cirt.net/code/nikto.shtml">www.cirt.net/code/nikto.shtml</a>
ScanDo	<a href="http://www.kavado.com">www.kavado.com</a>
WebEnforcer	<a href="http://www.hp.com/security/webenforcer">www.hp.com/security/webenforcer</a>
WebInspect	<a href="http://www.spidynamics.com">www.spidynamics.com</a>
Web Scarab	<a href="http://www.owasp.com">www.owasp.com</a>
WebSleuth	<a href="http://www.geocities.com/dzzie/sleuth/">www.geocities.com/dzzie/sleuth/</a>
Whisker and LibWhisker	<a href="http://www.wiretrip.net">www.wiretrip.net</a>
WhiteHat Arsenal	<a href="http://community.whitehatsec.com/wharsenal/">http://community.whitehatsec.com/wharsenal/</a>

### Database ASATs

AppDetective	<a href="http://www.appsecinc.com">www.appsecinc.com</a>
Database Scanner	<a href="http://www.iss.net">www.iss.net</a>
OraScan	<a href="http://www.nextgenss.com">www.nextgenss.com</a>

### Developer ASATs

BFBTester	<a href="http://sourceforge.net/projects/bfbtester/">http://sourceforge.net/projects/bfbtester/</a>
CLint	<a href="http://sourceforge.net/projects/clint/">http://sourceforge.net/projects/clint/</a>
Cqual	<a href="http://www.cs.berkeley.edu/~jfooster/cqual/">www.cs.berkeley.edu/~jfooster/cqual/</a>
FlawFinder	<a href="http://www.dwheeler.com/flawfinder/">www.dwheeler.com/flawfinder/</a>
Fuzz	<a href="http://sourceforge.net/projects/fuzz/">http://sourceforge.net/projects/fuzz/</a>
ITS4	<a href="http://www.cigital.com/its4/">www.cigital.com/its4/</a>
Jlint	<a href="http://artho.com/jlint">http://artho.com/jlint</a>
PyChecker	<a href="http://pychecker.sourceforge.net/">http://pychecker.sourceforge.net/</a>
RATS	<a href="http://www.securesw.com/rats.php">www.securesw.com/rats.php</a>
Splint	<a href="http://lclint.cs.virginia.edu/">http://lclint.cs.virginia.edu/</a>

### General Purpose ASATs

Bv-Control	<a href="http://www.bindview.com">www.bindview.com</a>
FoundScan	<a href="http://www.foundstone.com">www.foundstone.com</a>
Nessus	<a href="http://www.nessus.org">www.nessus.org</a>
NetRecon	<a href="http://enterprisecurity.symantec.com/products">http://enterprisecurity.symantec.com/products</a>
Retina	<a href="http://www.eeye.com">www.eeye.com</a>
SAINT	<a href="http://www.saintcorporation.com">www.saintcorporation.com</a>
Security Analyzer	<a href="http://www.netiq.com/products/sa/">www.netiq.com/products/sa/</a>
STAT Scanner	<a href="http://www.statonline.com">www.statonline.com</a>
Typhon II	<a href="http://www.nextgenss.com">www.nextgenss.com</a>

### Other Useful Tools

## FOR INFORMATION PURPOSE

## Draft

Achilles	<a href="http://www.digizen-security.com">www.digizen-security.com</a>
AppShield	<a href="http://www.sanctuminc.com">www.sanctuminc.com</a>
Blast	<a href="http://www.foundstone.com/knowledge/proddesc/blast.html">www.foundstone.com/knowledge/proddesc/blast.html</a>
IDA Pro Disassembler	<a href="http://www.datarescue.com/idabase/">www.datarescue.com/idabase/</a>
Metis	<a href="http://www.severus.org/sacha/metis/">www.severus.org/sacha/metis/</a>
PatchLink Update	<a href="http://www.patchlink.com">www.patchlink.com</a>
SecurityExpressions	<a href="http://www.pedestalsoftware.com">www.pedestalsoftware.com</a>
STAT Analyzer	<a href="http://www.statonline.com">www.statonline.com</a>
STAT Neutralizer	<a href="http://www.statonline.com">www.statonline.com</a>
StormFrontand	<a href="http://www.okena.com">www.okena.com</a>
StormWatch	
WebProxy	<a href="http://www.@stake.com/research/tools">www.@stake.com/research/tools</a>