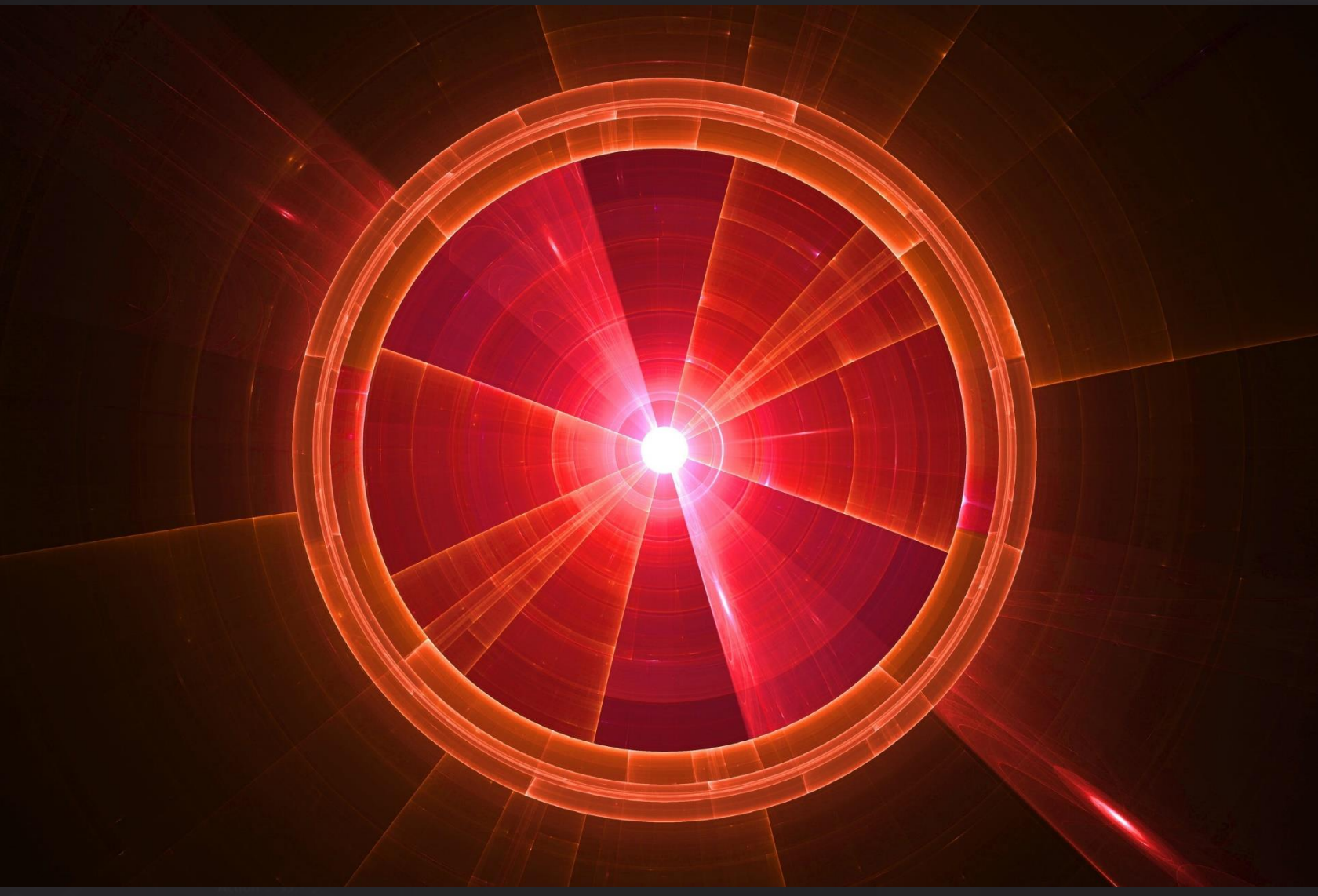
GRAMMATECH

DEFENSE INFORMATION SYSTEMS AGENCY(DISA) APPLICATION SECURITY AND DEVELOPMENT SECURITY TECHNICAL IMPLEMENTATION GUIDE (STIG) FINDING IDS

MAPPED TO CODESONAR® 6.2 WARNING CLASSES



TRUSTED LEADERS OF SOFTWARE ASSURANCE AND ADVANCED CYBER-SECURITY SOLUTIONS

WWW.GRAMMATECH.COM

INTRODUCTION

CodeSonar supports checking for violations of some of the requirements laid out in the DISA Ap- plication Security and Development STIG.

In particular, CodeSonar 6.2p0 provides mappings between CodeSonar warning classes and Finding IDs for two versions of this STIG: Version 4 Release 3 (release date April 28, 2017) and Version3 Release 10 (release date January 23, 2015).

Every CodeSonar warning report includes any Finding IDs from these versions that are closely mapped to the warning's class. (The close mapping for a warning class is the set of categories—including Application Security and Development Security STIG Finding IDs—that most closely matchthe class, if any).

You can configure CodeSonar to enable and disable warning classes mapped to specific Finding IDs from either or both of these versions, or use build presets to enable all warning classes that areclosely mapped to any Finding ID from one or both versions. In addition, you can use the CodeSonar search function to find warnings related to specific Finding IDs, or to any Finding ID from one or both versions.

We also provide broad mappings for these two STIG versions. The broad mapping for ApplicationSecurity and Development Security STIG and a given warning class includes the close mapping forthe class, plus any other Application Security and Development Security STIG finding IDs that arerelated to the class in a meaningful way, but not eligible for the close mapping.

This document contains four tables showing mappings between CodeSonar v6.2p0 warning classes and Finding IDs from the Application Security and Development Security STIG.

- Mappings for the DISA Application Security and Development Security STIG version 4release 3 (v4r3).
  - Close DISA Application Security and Development STIG v4r3 Mappings(CodeSonar v6.2p0)
  - Broad DISA Application Security and Development STIG v4r3 Mappings(CodeSonar v6.2p0)
- Mappings for the DISA Application Security and Development Security STIG version 3release 10 (v3r10).
  - Close DISA Application Security and Development STIG v3r10 Mappings(CodeSonar v6.2p0)
  - Broad DISA Application Security and Development STIG v3r10 mappings(CodeSonar v6.2p0)

For more information on the DISA Application Security and Development Security

STIG:https://iase.disa.mil/stigs/app-security/app-security/Pages/app-security.aspx

GrammaTech is a leading global provider of application testing (AST) solutions used by the world'smost security conscious organizations to detect, measure, analyze and resolve vulnerabilitiesfor software they develop or use. The company is also a trusted cybersecurity and artificial intelligence research partner for the nation's civil, defense, and intelligence agencies.

CLOSE DISA APPLICATION SECURITY AND DEVELOPMENT STIG V4R3 MAPPINGS(CODESONAR V6.2P0)

The following table contains CodeSonar warning classes that are closely mapped to Finding IDs from version 4, release 3 of the DISA Application Security and Development Security STIG. weakness IDs.

DISA Application Security and Development STIG(Version 4, Release 3) Finding ID and Title		CodeSonar Warning Class Name
V-69257	The application must implement DoD-approved encryption to protect the confidentiality of remoteaccess sessions.	Encryption without Padding
V-69259	The application must implement cryptographicmechanisms to protect the integrity of remote access sessions.	Encryption without padding Use of crypt Use of rand Use of rand48 function Use of random Weak Cryptography
V-69567	The application must only store cryptographicrepresentations of passwords.	Plaintext Storage of Password Plaintext Transmission of Password
V-69569	The application must transmit only cryptographically-protected passwords.	Plaintext Storage of Password
V-70185	The application must not be vulnerable to raceconditions.	Data Race File System Race Condition
V-70191	The application must utilize FIPS-validated cryptographic modules when signing applicationcomponents.	Use of crypt Use of rand Use of rand48 Function Use of random Weak Cryptography
V-70193	The application must utilize FIPS-validated cryptographic modules when generating cryptographic hashes.	Use of crypt Use of rand Use of rand48 Function Use of random Weak Cryptography
V-70195	The application must utilize FIPS-validated cryptographic modules when protecting unclassified information that requires cryptographic protection.	Use of crypt Use of rand Use of rand48 Function Use of random Weak Cryptography
V-70217	The application must use the Federal Information Processing Standard (FIPS) 140-2-validated cryptographic modules and random number generatorif the application implements encryption, key exchange, digital signature, and hash functionality.	Use of crypt Use of rand Use of rand48 Function Use of random Weak Cryptography

V-70229	The application must use appropriate cryptography in order to protect stored DoD informationwhen required by the information owner or DoDpolicy.	Encryption without Padding Use of crypt Weak Cryptography
V-70245	The application must protect the confidentiality and integrity of transmitted information.	Encryption without Padding
V-70257	The application must protect from Cross-Site Scripting (XSS) vulnerabilities.	Cross Site Scripting
V-70261	The application must protect from command injection.	Command Injection Library Injection Untrusted Library Load Untrusted Process Creation Use of execlp Use of xecvp Use of popen Use of system
V-70265	The application must validate all input.	Command Injection Format String Injection LDAP Injection Library Injection SQL Injection Tainted Allocation Size Tainted Buffer Access Tainted Configuration SettingTainted Network Address Tainted Write Untrusted Library Load Untrusted Network Host Untrusted Network Port Untrusted Process Creation
V-70267	The application must not be vulnerable to SQLInjection.	SQL Injection
V-70269	The application must not be vulnerable to XML-oriented attacks.	Insecure XSLT Execution Possible XML External Entity Reference Tainted XML

V-70271	The application must not be subject to input han-dling vulnerabilities.	Format String Injection LDAP Injection Library Injection SQL Injection Tainted Allocation Size Tainted Buffer Access Tainted Configuration Setting Tainted Network Address Tainted Write Untrusted Library Load Untrusted Network Host Untrusted Network Port

V-70277	The application must not be vulnerable to overflow attacks.	Addition Overflow of Allocation Size Addition Overflow of Size Buffer Overrun Buffer Underrun File System Race Condition Format String Format String Injection Hardcoded DNS Name Inappropriate Character Arithmetic Integer Overflow of Allocation Size Multiplication Overflow of Allocation Size Multiplication Overflow of Size No Space For Null Terminator Subtraction Underflow of Allocation Size Subtraction Underflow of Size Tainted Buffer Access Tainted Filename Unreasonable Size Argument Use of getopt Use of getpass Use of gets Use of getwd Use of OemToAnsi Use of OemToChar Use of realpath Use of recvmsg Use of strcat Use of StrCatChainW Use of strcmp Use of strcpy Use of strlen Use of strtrns Use of syslog Use of strstr Use of strpbrk Use of strrchr Use of strchr Use of strtok Use of strspn Use of strcspn Cast Alters Value Coercion Alters Value Risky Integer Promotion
V-70363	The application must not contain embeddedauthentication data.	Hardcoded Authentication
V-70391	The application must not be subject to error han-dling vulnerabilities.	Ignored Return Value

BROAD DISA APPLICATION SECURITY AND DEVELOPMENT STIG V4R3 MAPPINGS(CODESONAR V6.2P0)

The following table contains CodeSonar warning classes that are broadly mapped to Finding IDs from version 4, release 3 of the DISA Application Security and Development Security STIG.

Warning classes that are also in the close mapping are in bold text.

DISA Application Security and Development STIG (Version 4, release 3) Finding ID and Title		**Broadly Mapped CodeSonar Warning Classes**
V-69257	The application must implement DoD-approved encryption to protect the confidentiality of remote access sessions.	**Encryption without Padding**
V-69259	The application must implement cryptographic mechanisms to protect the integrity of remote access sessions.	**Encryption without Padding** **Use of crypt** **Use of rand** **Use of rand48 Function** **Use of random** **Weak Cryptography**
V-69567	The application must only store cryptographic representations of passwords.	**Plaintext Storage of Password** **Plaintext Transmission of Password**
V-69569	The application must transmit only cryptographically-protected passwords.	**Plaintext Storage of Password** **Plaintext Transmission of Password**
V-70185	The application must not be vulnerable to race conditions.	**Data Race** **File System Race Condition** **Multiple Accesses of Atomic**
V-70191	The application must utilize FIPS-validated cryptographic modules when signing application components.	**Use of crypt** **Use of rand** **Use of rand48 Function** **Use of random** **Weak Cryptography**
V-70193	The application must utilize FIPS-validated cryptographic modules when generating cryptographic hashes.	**Use of crypt** **Use of rand** **Use of rand48 Function** **Use of random** **Weak Cryptography**
V-70195	The application must utilize FIPS-validated cryptographic modules when protecting unclassified information that requires cryptographic protection.	**Use of crypt** **Use of rand** **Use of rand48 Function** **Use of random** **Weak Cryptography**

V-70217	The application must use the Federal Information Processing Standard (FIPS) 140-2-validated cryptographic modules and random number generator if the application implements encryption, key exchange, digital signature, and hash functionality.	**Use of crypt** **Weak Cryptography**
V-70229	The application must use appropriate cryptography in order to protect stored DoD information when required by the information owner or DoD policy.	**Encryption without Padding** **Use of crypt** **Weak Cryptography**
V-70245	The application must protect the confidentiality and integrity of transmitted information.	**Encryption without Padding**
V-70257	The application must protect from Cross-Site Scripting (XSS) vulnerabilities.	**Cross Site Scripting**
V-70261	The application must protect from command injection.	**Command Injection** **Library Injection** **Untrusted Library Load** **Untrusted Process Creation** **Use of execlp** **Use of execvp** **Use of popen** **Use of system**
V-70265	The application must validate all input.	**Command Injection** **Format String Injection** **LDAP Injection** **Library Injection** **SQL Injection** **Tainted Allocation Size** **Tainted Buffer Access** **Tainted Configuration Setting** **Tainted Network Address** **Tainted Write** **Untrusted Library Load** **Untrusted Network Host** **Untrusted Network Port** **Untrusted Process Creation**
V-70267	The application must not be vulnerable to SQL Injection.	**SQL Injection**
V-70269	The application must not be vulnerable to XML-oriented attacks	**Insecure XSLT Execution** **Possible XML External Entity Reference** **Tainted XML**

V-70271	The application must not be subject to input handling vulnerabilities.	**Format String Injection** **LDAP Injection** **Library Injection** **SQL Injection** **Tainted Allocation Size** **Tainted Buffer Access** **Tainted Configuration Setting** **Tainted Network Address** **Tainted Write** **Untrusted Library Load** **Untrusted Network Host** **Untrusted Network Port**

V-70277	The application must not be vulnerable to overflow attacks.	Basic Numerical Type UsedCast Alters Value Coercion Alters ValueDivision By Zero Expression Value Widenedby Assignment Expression Value Widenedby Other Operand Float Division By Zero Inappropriate AssignmentType Inappropriate Bit-field Type Inappropriate Cast Type Inappropriate Cast Type:Expression Inappropriate Operand TypeNegative Character Value Negative Shift Amount Risky Integer Promotion Shift Amount Exceeds BitWidth **Addition Overflow ofAllocation Size** **Addition Overflow of SizeBuffer Overrun** **Buffer Underrun** **File System RaceCondition** **Format String** **Format String Injection Hardcoded** **DNS Name** **Inappropriate CharacterArithmetic** **Integer Overflow ofAllocation Size** **Multiplication Overflow ofAllocation Size** **Multiplication Overflowof Size** **No Space For NullTerminator** **Subtraction Underflow of Allocation Size** **Subtraction Underflowof Size** **Tainted Buffer AccessUse of strstr** **Use of strpbrk**	**Use of strrchr** **Use of strchr** **Use of strtok** **Use of strspn** **Use of strcspn** **Cast Alters Value** **Coercion Alters Value** **Risky Integer Promotion** Dynamic Invalid Read (CodeSonar/X only) Dynamic Invalid Write (CodeSonar/X only) **Tainted Filename** **Unreasonable Size Argument** **Use of getopt** **Use of getpass** **Use of gets** **Use of getwd** **Use of OemToAnsi** **Use of OemToChar** **Use of realpath** **Use of recvmsg** **Use of strcat** **Use of StrCatChainW** **Use of strcmp** **Use of strcpy** **Use of strlen** **Use of strtrns** **Use of syslog**
V-70363	The application must not contain embedded authenticationdata.	**Hardcoded Authentication**	
V-70391	The application must not be subject to error handling vulnera-bilities.	**Ignored Return Value**	

V-70403	Default passwords must be changed.	Hardcoded Password Password in Property File

CLOSE DISA APPLICATION SECURITY AND DEVELOPMENT STIG V3R10 MAPPINGS(CODESONAR V6.2P0)

The following table contains CodeSonar warning classes that are closely mapped to Finding IDsfrom version 3, release 10 of the DISA Application Security and Development Security STIG.

DISA Application Security and Development STIG (Version 3, release 10) Finding ID and Title		**Closely Mapped CodeSonar Warning Classes**
V-6135	The designer will ensure the appropriate cryptography is used to protect stored DoD information if required by the informationowner.	Encryption without Padding Use of crypt
V-6136	The designer will ensure data transmitted through a commer- cial or wireless network is protected using an appropriate formof cryptography.	Encryption without Padding
V-6137	The designer will ensure the application uses the Federal Information Processing Standard (FIPS) 140-2 validated cryp- tographic modules and random number generator if the appli-cation implements encryption, key exchange, digital signature,and hash functionality.	Use of crypt Use of rand Use of rand48 Function Use of random
V-6149	The designer will ensure the application does not contain source code that is never invoked during operation, except forsoftware components and libraries from approved third-party products.	Unexercised Call Unexercised Computation Unexercised Conditional Unexercised Control Flow Unexercised Data Flow Unreachable Call Unreachable Computation Unreachable Conditional Unreachable Control Flow Unreachable Data Flow Unused Label Unused Macro Unused Parameter Unused Tag Unused Type Unused Value Unused Variable
V-6156	The designer will ensure the application does not contain em-bedded authentication data.	Hardcoded Authentication

V-6157	The designer will ensure the application does not contain inval-id URL or path references.	Dangerous Include File Name LDAP Injection Tainted Filename Tainted Network Address Use of _exec Use of _spawn Use of AfxLoadLibrary Use of CoLoadLibrary Use of CreateProcess Use of execlp Use of execvp Use of LoadLibrary Use of LoadModule Use of MoveFile Use of popen Use of SHCreateProcessAsUserW Use of ShellExecute Use of system Use of t_open Use of WinExec
V-6164	The designer will ensure the application validates all input.	Command Injection Format String Injection LDAP Injection Library Injection SQL Injection Tainted Allocation Size Tainted Buffer Access Tainted Configuration SettingTainted Network Address Tainted Write

V-6165	The designer will ensure the application does not have buffer overflows, use functions known to be vulnerable to buffer over-flows, and does not use signed values for memory allocation where permitted by the programming language.	Buffer Overrun Buffer Underrun No Space For Null Terminator Tainted Buffer Access Unreasonable Size Argument Use of getopt Use of getpass Use of gets Use of getwd Use of OemToAnsi Use of OemToChar Use of realpath Use of recvmsg Use of strcat Use of StrCatChainW Use of strcmp Use of strcpy Use of strlen Use of strtrns Use of syslog Use of strstr Use of strpbrk Use of strrchr Use of strchr Use of strtok Use of strspn Use of strcspn
V-6166	The designer will ensure the application is not subject to errorhandling vulnerabilities.	Ignored Return Value
V-16793	The designer will ensure the application properly clears or overwrites all memory blocks used to process sensitive data, ifrequired by the information owner, and clears or overwrites all memory blocks used for classified data.	Use of memset
V-16796	The designer will ensure the application transmits accountpasswords in an approved encrypted format.	Plaintext Storage of Password Plaintext Transmission of Password
V-16797	The designer will ensure the application stores account pass-words in an approved encrypted format.	Plaintext Storage of Password Plaintext Transmission of Password
V-16804	The designer will ensure the application does not rely solely ona resource name to control access to a resource.	File System Race Condition Hardcoded DNS Name Tainted Filename

V-16807	The designer will ensure the application is not vulnerable to SQL Injection, uses prepared or parameterized statements, does not use concatenation or replacement to build SQL que-ries, and does not directly access the tables in a database.	SQL Injection
V-16808	The designer will ensure the application is not vulnerable tointeger arithmetic issues.	Addition Overflow of Allocation Size Addition Overflow of Size Inappropriate Character Arithmetic Integer Overflow of Allocation Size Multiplication Overflow of Allocation Size Multiplication Overflow of Size Subtraction Underflow of Allocation Size Subtraction Underflow of Size Cast Alters Value Coercion Alters Value Division By Zero Negative Shift Amount Risky Integer Promotion
V-16809	The designer will ensure the application does not contain for-mat string vulnerabilities.	Format String Format String Injection
V-16810	The designer will ensure the application does not allow com-mand injection.	Command Injection Library Injection Use of execlp Use of execvp Use of popen Use of system
V-16815	The designer will ensure the application is not vulnerable torace conditions.	Data Race File System Race Condition

BROAD DISA APPLICATION SECURITY AND DEVELOPMENT STIG V3R10 MAPPINGS(CODESONAR V6.2P0)

The following table contains CodeSonar warning classes that are broadly mapped to Finding IDsfrom version 3, release 10 of the DISA Application Security and Development Security STIG.

Warning classes that are also in the close mapping are in bold text.

DISA Application Security and Development STIG (Version 3, release 10) Finding ID and Title		**Broadly Mapped CodeSonar Classes**
V-6135	The designer will ensure the appropriate cryptography is used to pro-tect stored DoD information if required by the information owner.	**Encryption without Padding** **Use of crypt**
V-6136	The designer will ensure data transmitted through a commercial or wireless network is protected using an appropriate form of cryptogra-phy.	**Encryption without Padding**
V-6137	The designer will ensure the application uses the Federal InformationProcessing Standard (FIPS) 140-2 validated cryptographic modules and random number generator if the application implements encryp-tion, key exchange, digital signature, and hash functionality.	**Use of crypt** **Use of rand** **Use of rand48 Function** **Use of random**
V-6149	The designer will ensure the application does not contain source codethat is never invoked during operation, except for software compo- nents and libraries from approved third-party products.	**Unexercised Call** **Unexercised Computation** **Unexercised Conditional** **Unexercised Control Flow** **Unexercised Data Flow** **Unreachable Call** **Unreachable Computation** **Unreachable Conditional** **Unreachable Control Flow** **Unreachable Data Flow** **Unused Label** **Unused Macro** **Unused Parameter** **Unused Tag** **Unused Type** **Unused Value** **Unused Variable**
V-6156	The designer will ensure the application does not contain embeddedauthentication data.	**Hardcoded Authentication**

V-6157	The designer will ensure the application does not contain invalid URL orpath references.	**Dangerous Include File Name** **LDAP Injection** **Tainted Filename** **Tainted Network Address** **Use of _exec** **Use of _spawn** **Use of AfxLoadLibrary** **Use of CoLoadLibrary** **Use of CreateProcess** **Use of execlp** **Use of execvp** **Use of LoadLibrary** **Use of LoadModule** **Use of MoveFile** **Use of popen** **Use of SHCreateProcessAsUserW** **Use of ShellExecute** **Use of system** **Use of t_open** **Use of WinExec**
V-6164	The designer will ensure the application validates all input.	**Command Injection** **Format String Injection** **LDAP Injection** **Library Injection** **SQL Injection** **Tainted Allocation Size** **Tainted Buffer Access** **Tainted Configuration Setting** **Tainted Network Address** **Tainted Write**

V-6165	The designer will ensure the application does not have buffer overflows,use functions known to be vulnerable to buffer overflows, and does not use signed values for memory allocation where permitted by theprogramming language.	Addition Overflow of Allocation Size Integer Overflow of Allocation Size Multiplication Overflow of Allocation Size Subtraction Underflow of Allocation Size **Buffer Overrun** **Buffer Underrun** **No Space For Null Terminator** **Tainted Buffer Access** **Unreasonable Size Argument** **Use of getopt** **Use of getpass** **Use of gets** **Use of getwd** **Use of OemToAnsi** **Use of OemToChar** **Use of realpath** **Use of recvmsg** **Use of strcat** **Use of StrCatChainW** **Use of strcmp** **Use of strcpy** **Use of strlen** **Use of strtrns** **Use of syslog** **Use of strstr** **Use of strpbrk** **Use of strrchr** **Use of strchr** **Use of strtok** **Use of strspn** **Use of strcspn**
V-6166	The designer will ensure the application is not subject to error handlingvulnerabilities.	**Ignored Return Value**
V-16793	The designer will ensure the application properly clears or overwrites all memory blocks used to process sensitive data, if required by the information owner, and clears or overwrites all memory blocks used forclassified data.	**Use of memset**
V-16796	The designer will ensure the application transmits account passwordsin an approved encrypted format.	**Plaintext Storage of Password** **Plaintext Transmission of Password**
V-16797	The designer will ensure the application stores account passwords inan approved encrypted format.	**Plaintext Storage of Password** **Plaintext Transmission of Password**

V-16804	The designer will ensure the application does not rely solely on a re-source name to control access to a resource.	**File System Race Condition** **Hardcoded DNS Name** **Tainted Filename**
V-16807	The designer will ensure the application is not vulnerable to SQL Injec-tion, uses prepared or parameterized statements, does not use con- catenation or replacement to build SQL queries, and does not directlyaccess the tables in a database.	**SQL Injection**
V-16808	The designer will ensure the application is not vulnerable to integerarithmetic issues.	**Basic Numerical Type Used** **Cast Alters Value** **Coercion Alters Value** **Division By Zero** **Expression Value Widened by Assignment** **Expression Value Widened by Other Operand** **Inappropriate Assignment Type** **Inappropriate Bit-field Type** **Inappropriate Cast Type** **Inappropriate Cast Type: Expression** **Inappropriate Operand Type** **Negative Character Value** **Negative Shift Amount** **Risky Integer Promotion** **Shift Amount Exceeds Bit Width** **Addition Overflow of Allocation Size** **Addition Overflow of Size** **Inappropriate Character** **Arithmetic Integer Overflow of Allocation Size** **Multiplication Overflow of AllocationSize** **Multiplication Overflow of Size** **Subtraction Underflow of AllocationSize** **Subtraction Underflow of Size**
V-16809	The designer will ensure the application does not contain format stringvulnerabilities.	**Format String** **Format String Injection**

V-16810	The designer will ensure the application does not allow commandinjection.	**Command Injection** **Library Injection** **Use of execlp** **Use of execvp** **Use of popen** **Use of system**
V-16815	The designer will ensure the application is not vulnerable to race con-ditions.	**Data Race** **File System Race Condition** Multiple Accesses of Atomic