

Threat Assessment and Remediation Analysis (TARA) Overview

October 2013



Approved for Public
Release: 11-4987.

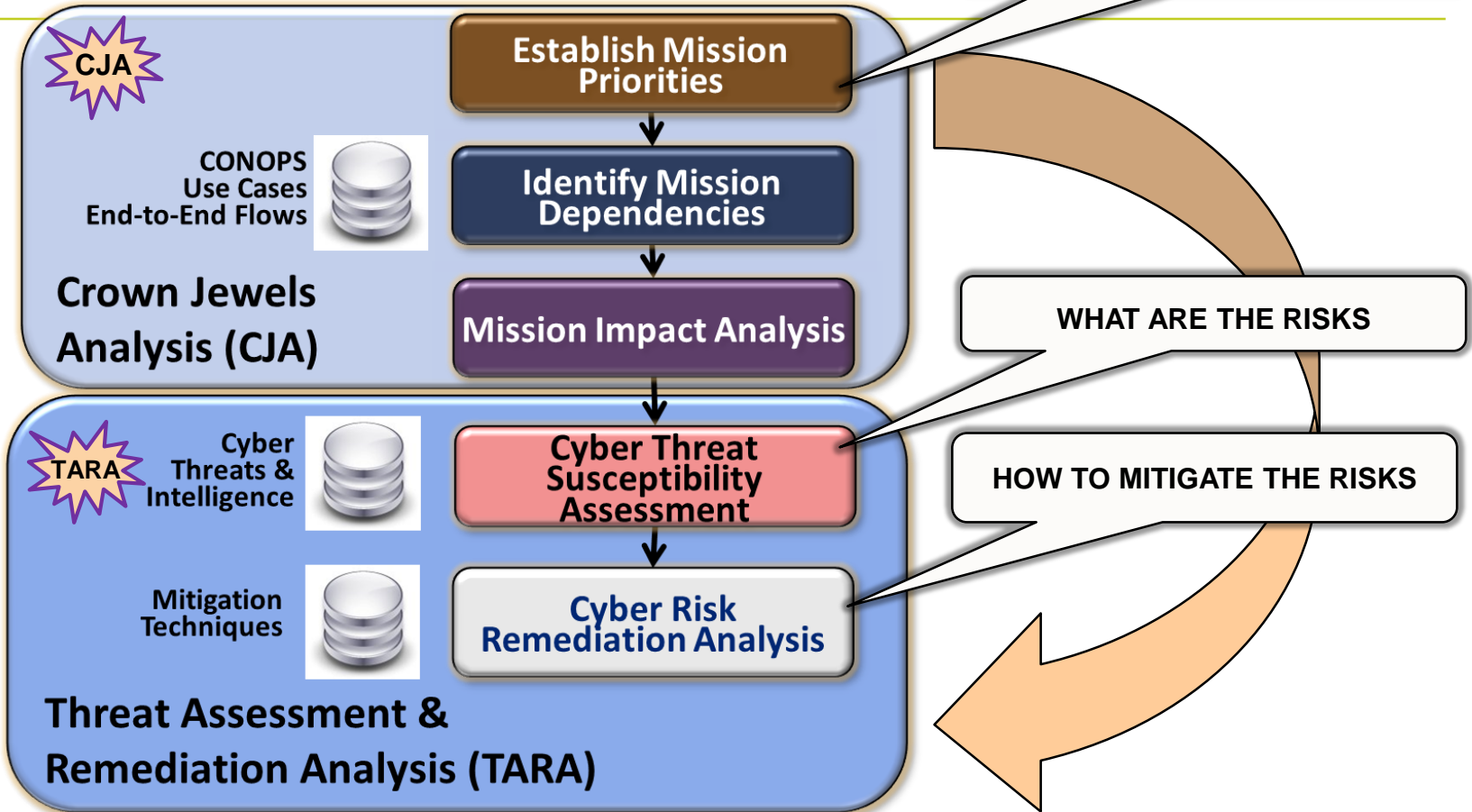
Distribution Unlimited.

Threat Assessment & Remediation Analysis (TARA)

- **Methodology to identify and assess cyber threats and select countermeasures effective at mitigating those threats**
 - Leverages catalog of Attack Vectors (AVs), Countermeasures (CMs), and associated mappings
 - Use of catalog ensures that findings are consistent across assessments
 - Uses scoring models to quantitatively assess AVs and CMs
 - AVs ranked by risk, providing a basis for effective triage
 - CMs ranked by cost-effectiveness, providing a basis for identifying optimal solutions
 - Delivers recommendations
 - Allows programs to make informed choices on how best to improve a system's security posture and resilience
 - Can be performed standalone or as follow-on to criticality or mission impact (MI) analysis, such as Crown Jewels Analysis (CJA)
 - TARA performed in tandem with CJA supports Mission Assurance Engineering (MAE) objectives

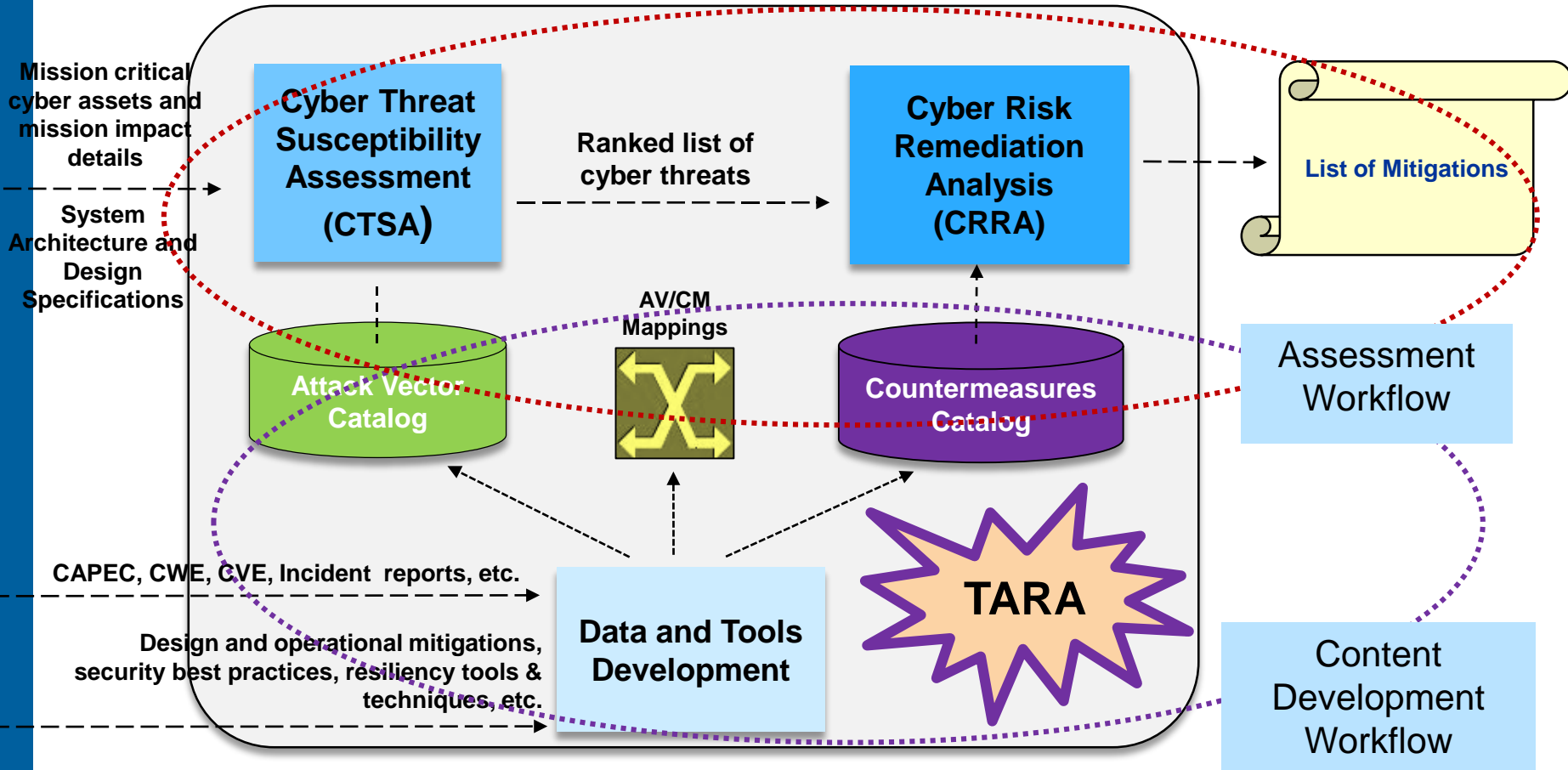
Mission Assurance Engineering (MAE)

The Big Picture



CJA and TARA together support the identification, assessment, and mitigation of cyber risk to mission essential assets

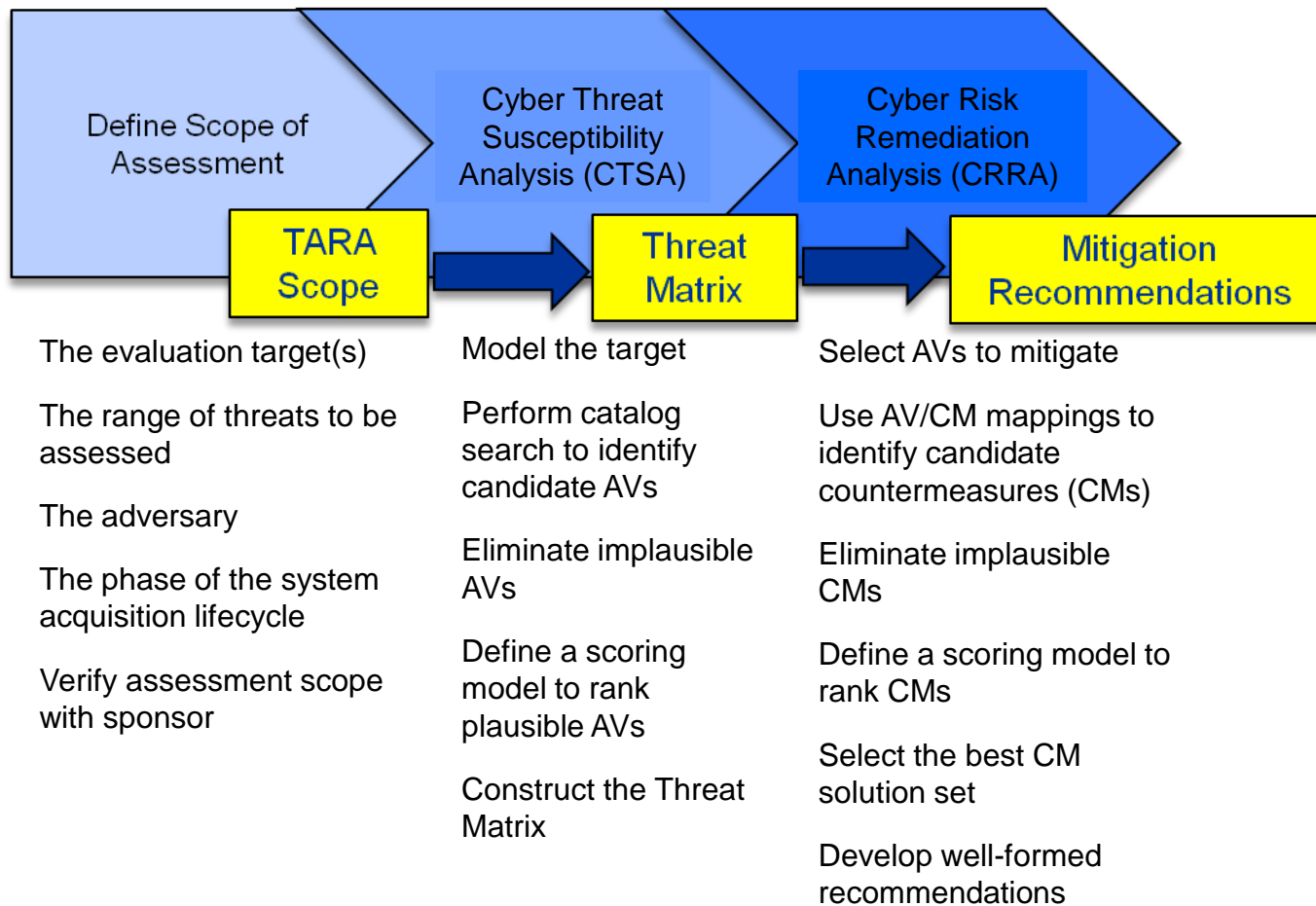
TARA Methodology Workflows



Workflow – Sequence of connected activities that produce useful work

TARA Assessment Workflow

Objective to identify and assess cyber threats and select countermeasures effective at mitigating those threats



Approved for Public Release

TARA Assessment Products

Threat Susceptibility Matrix

ID	Attack Vector Name	Risk Score	Evaluation Targets		
			Router	Web Server	Browser
T000105	Cross Site Scripting	2.1			X
T000008	Unsecured SNMP agent	1.9	X		
T000016	Simple Script Injection	1.8	X		X
T000049	Buffer Overflow	1.7	X	X	X
T000001	BIOS replaced with version that allows unsigned updates	1.6	X	X	X
T000021	Man in the Middle Attack	1.4		X	X

Provides a ranked list of cyber threats, mapped to components of the evaluation target

Answers the questions: Where and how is my system most susceptible?

Solution Effectiveness Table

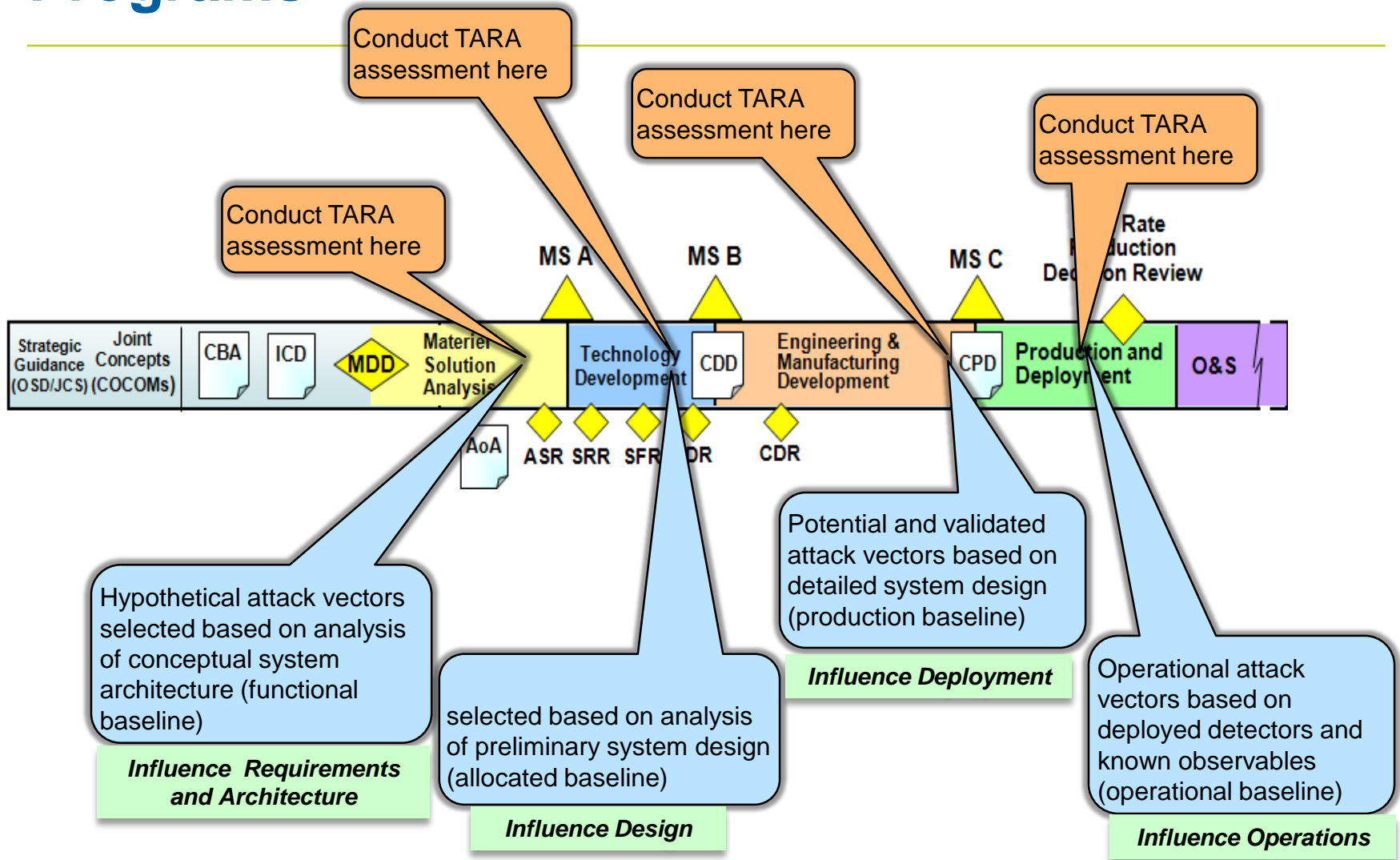
System name: System XYZ				Assurance Level: Medium				
Countermeasure (CM)			Mitigation Effectiveness (by Attack Vector ID)					
ID	Countermeasure Name	Cost Index	T000105	T000008	T000016	T000049	T000001	T000021
			2.1	1.9	1.8	1.7	1.6	1.4
C000023	Change default SNMP community string values	1		P				
C000062	Disable client side scripting	3	P		P			
C000194	Disable hyperlinks in email	1	M		M			
C000015	Verify BIOS implemented security controls after BIOS image update	2					P	
C000018	Use checksums to verify the integrity of downloaded BIOS image updates	2					P	
C000024	Restrict SNMP community string value reuse	2		P				
C000081	Use strong mutual authentication	3						P
C000083	Use cryptography that is sufficient strong	3						P
C000136	Utilize processor-based protection capabilities	1				M		
C000238	Enforce software quality standards and guidelines that improve software quality	2				M		
C000090	Validate input fields use of NULL, escape, backslash, meta, and control characters	3	M		M			
C000002	Verify BIOS image write protection	2					M	
C000101	Verify buffer sizes	2				M		
C000247	Ensure trustworthiness of key personnel	3						M
Totals		30	3	2	3	3	3	3

Provides a ranked list of countermeasures, mapped to cyber threats, and identifies the preventative or mitigating effect each countermeasure provides

Answers the questions: How are my threats mitigated and where are the gaps?

Approved for Public Release

Threat-based Analysis Influence on Acquisition Programs



Approved for Public Release

© 2013 The MITRE Corporation. All rights reserved

MITRE

Applications of TARA

- **Threat-based Analysis of System Architecture**
- **Systems Security Engineering (SSE)**
- **Support to Acquisition Programs**
- **Program Protection Planning**
- **Vulnerability Assessment Planning**
- **Supply Chain Risk Management (SCRM) Analysis**

TARA

Catalog and Toolset

Approved for Public Release

© 2013 The MITRE Corporation. All rights reserved

MITRE

Objectives of the TARA Catalog



- Provide a repository of Attack Vector (AV) and Countermeasure (CM) data used in TARA assessments
- Implement a standard data model to represent AVs and CMs
- Help establish consistency from one TARA assessment to the next

TARA Catalog Data

Vector Groups (VGs)

Attack Vectors (AVs)

Countermeasures (CMs)

AC ID	AC Name	Keywords
A000223	Applications	antivirus browser excel MS project MS word Outlook pdf reader powerpoint visio vpn internet explorer firefox
A000036	authentication	credential password account authentication certificate username authenticate user SAML token credentials
A000182	Data	DOM html parse schema Unicode XHTML XML cookie token
A000032	database	database Oracle SQL schema DBMS JDBC MS access ODBC
A000201	email	email IMAP POP SMTP Outlook Thunderbird
A000052	firmware	BIOS firmware IOS
A000267	mobile	3G 4G 802.11 access point cell cellular hotspot mobile WEP wi-fi wiimax wireless WPA
A000058	network service	IDS IPS proxy
A000235	OS	android IOS linux OS unix windows
A000128	OSI - Application Layer	BGP DHCP DNS FTP http HTTPS IMAP LDAP POP SIP SMTP SNMP SSL
A000140	OSI - Data Link Layer	ARP OSPF VLAN
A000136	OSI - Network Layer	ICMP IP IPv4 IPv6
A000131	OSI - Transport Layer	TCP UDP
A000251	PKI	certificate CRL keystore PKI revocation root self-signed X.509 X509 CA certificate authority
A000051	platform	bridge cloud firewall gateway hub router server switch thick client thin client wireless
A000228	Remote access	IPsec SSH telnet vpn
A000129	Scripting	CGI JavaScript Perl PHP Python flash bash
A000172	Security	access matrix ACL AES biometric certificate CHAP DES digital signature EAP encryption firewall hash IPsec kerberos L2F L2TP MD5 packet filter password PKI PPTP radius security SHA SSH

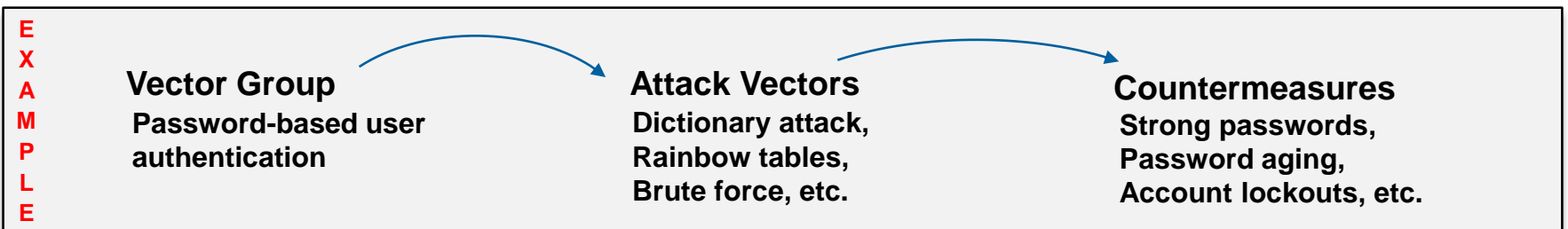
Named collection of attack vectors, e.g., architectural components, technologies, shopping carts, intrusion sets etc.

TTP ID	TTP Name
T000001	Malicious BIOS code allows unsigned updates
T000002	Secure BIOS update bypassed via buffer overflow
T000003	User installs malicious BIOS image on device
T000004	Malware reflashes device with malicious BIOS
T000005	System is rolled back to an authentic but vulnerable system BIOS
T000006	Compromised update server distributes malicious BIOS
T000007	SNMP community strings transmitted in the clear
T000008	SNMP Community String Name is Guessable
T000009	Session Credential Falsification through Prediction
T000010	HTTP Request Smuggling
T000011	Lifting Data Embedded in Client Distributions
T000012	Postfix, Null Terminate, and Backslash
T000013	Exploiting Trust in Client
T000014	Accessing, Intercepting, and Modifying HTTP Cookies
T000015	Cross Site Request Forgery (Session Riding)
T000016	Simple Script Injection
T000017	Subvert Code-signing Facilities
T000018	Using Unicode Encoding to Bypass Validation Logic
T000019	Using Escaped Slashes in Alternate Encoding
T000020	Xquery Injection
T000021	Man in the Middle Attack
T000022	Cryptanalysis
T000023	Cross Site Tracing
T000024	Malicious Software Update
T000026	Accessing Functionality Not Properly Constrained by ACLs
T000027	Manipulating Input to File System Calls

Adversary approaches to compromise a cyber asset

CM ID	CM Name
C000001	Verify secure BIOS update non-bypassability
C000002	Verify BIOS image write protection
C000003	Verify recovery process to restore last-known-good BIOS image
C000005	Institute secure BIOS update capabilities using RTU
C000006	Perform source code review of BIOS to identify software defects and potential vulnerabilities
C000007	Perform test and evaluation (TandE) of BIOS update mechanism
C000010	Restrict admin access to device
C000012	Enforce the 2-man rule when performing critical administrative functions
C000013	Conduct independent verification of software image once installed
C000015	Verify BIOS implemented security controls after BIOS image update
C000018	Use checksums to verify the integrity of downloaded BIOS image updates
C000020	Restrict access to the BIOS update server
C000021	Use latest version of SNMP protocol
C000022	Isolate network management traffic to internal network
C000023	Change default SNMP community string values
C000024	Restrict SNMP community string value reuse
C000025	Configure web servers to utilize strict parsing
C000027	Terminate client sessions after each request
C000028	Mark all sensitive web pages as non-cacheable
C000030	Conduct threat modeling
C000034	Reduce attack surface
C000039	Convert input data
C000041	Use same character encoding
C000045	Utilize high quality session IDs
C000047	Encrypt session cookies
C000049	Enforce client authentication
C000051	Use digital signatures

Approaches for mitigating attack vectors



Approved for Public Release

© 2013 The MITRE Corporation. All rights reserved

Attack Vectors (AVs)

“A sequence of steps performed by an adversary in the course of conducting a cyber attack”

■ Sources of Attack Vector data

- Open source data on attack patterns (CAPEC), software weaknesses (CWE), and vulnerabilities (CVE)
- NIST publications
- Details on security incidents that occur in the commercial sector
- Classified security incident reporting
- Security Threat Assessment Reports (STARs), Integrated Threat Assessments (ITAs), DIA Capstone, NASIC publications, etc.
- Published security research
 - Weaponized exploits detailed at Blackhat, Defcon, Schmooscon, etc.

Countermeasures (CMs)

“Actions, devices, procedures, or techniques that meet or oppose (i.e., counters) a threat, a vulnerability, or an attack by eliminating or preventing it, by minimizing the harm it can cause, or by discovering and reporting it so that corrective action can be taken.”

Source: CNSS 4009

■ Sources of CM data

- Open source data on attack patterns (CAPEC) and software weaknesses (CWE) often includes mitigation details
- NIST publications
- Industry recognized security best practices
- Published security research
 - Journal articles detailing new approaches for detecting anomalous behavior, malware, etc.

Vector Groups

A named collection of Attack Vectors

Types of Vector Groups

- Architectural groupings
 - Client server, network, hardware, software, API, etc.
- Technology groupings
 - Database, web service, XML, email, Unix, Windows, etc.
- Shopping carts
 - Handpicked collection of attack vectors used in an assessment

Vector Group	Group Details		
	Attack Vectors	Counter-measures	Created
<i>Network.routers</i>	34	32	4/1/12
<i>Network.firewalls</i>	7	13	10/15/11
<i>Malware</i>	9	38	10/15/11
<i>IdM.password</i>	8	15	10/15/11
<i>IdM.PKI</i>	6	14	9/1/11
<i>Webclient</i>	21	54	9/1/11
<i>Webservices.webserverplatform</i>	14	33	9/1/11
<i>Webservices.SOAP-UDDI-WSDL</i>	34	73	9/1/11
<i>Webservices.REST</i>			
<i>Webservices.HTTP-HTML-AJAX</i>			
<i>Virtualization</i>	7	12	4/1/12
<i>Crypto.SSL</i>	6	13	9/1/11
<i>Database</i>	6	22	9/1/11
<i>Messaging.JMS</i>	7	29	6/1/12
<i>XML</i>	7	17	2/1/12
<i>Supplychain.COTS</i>	11	52	9/1/11
<i>Software</i>	7	12	9/1/11
<i>Firmware.BIOS</i>	6	15	9/1/11
<i>IPNetwork.BGP</i>	8	17	1/15/13
<i>Networkmanagement.SNMP</i>	6	11	9/1/11
<i>GPS</i>	29	8	9/1/11
<i>Comms.Terrestrial</i>			
<i>Comms.LOS</i>			
<i>Comms.BLOS</i>			
<i>Comms.mobilewifi</i>	6	7	9/1/11

Example Vector Group: Software

Attack Vectors

T000005	Exploitation of a zero-day vulnerability
T000006	Counterfeit web sites used to distribute malicious software updates
T000009	Malicious software implantation through 3rd party bundling
T000010	Adversary gains unauthorized access by exploiting a software vulnerability
T000016	Unauthorized / unrestricted copying
T000017	Clandestine changes to software or mission data
T000021	Software defects hidden/obscured by code complexity

Entries are a partial listing, in no particular order

Countermeasures

C000003	Strip debug info from production executables
C000006	Establish a software pedigree
C000014	Make it difficult for the APT
C000019	Apply static code analysis tools to identify software defects
C000020	Establish coding guidelines to improve software quality
C000021	Select programming languages that minimize potential for software defects
C000022	Enforce configuration management (CM) practices that protect source code
C000025	Develop an assurance case for software
C000026	Use dynamic analysis tools to assess software for runtime defects
C000032	Perform risk assessments for open source and unsupported products
C000038	Design to log securely
C000043	Ensure that developers are trained in how to develop secure software

Sources include: Software Assurance Workforce Education and Training Working Group, "Software Assurance: A Curriculum Guide to the Common Body of Knowledge to Produce, Acquire and Sustain Secure Software", DHS, October 2007.

Approved for Public Release

TARA Toolset

Web-based tools supporting TARA assessments and catalog development

Catalog Search Tools

Filter	TTP ID	TTP Name
<input checked="" type="checkbox"/>	T000010	HTTP Request Smuggling
<input checked="" type="checkbox"/>	T000014	Accessing, Intercepting, and Modifying HTTP Cookies
<input checked="" type="checkbox"/>	T000016	Simple Script Injection
<input checked="" type="checkbox"/>	T000023	Cross Site Tracing
<input checked="" type="checkbox"/>	T000029	Exploitation of Session Variables, Resource IDs and other Trusted Credentials
<input checked="" type="checkbox"/>	T000066	Web Server/Application Fingerprinting
<input checked="" type="checkbox"/>	T000073	HTTP Response Splitting
<input checked="" type="checkbox"/>	T000076	HTTP Verb Tampering
<input checked="" type="checkbox"/>	T000078	Flash Parameter Injection
<input checked="" type="checkbox"/>	T000084	HTTP Response Smuggling
<input checked="" type="checkbox"/>	T000084	Web Logs Tampering
<input checked="" type="checkbox"/>	T000088	Modifying filename
<input checked="" type="checkbox"/>	T000096	Poison Web Service
<input checked="" type="checkbox"/>	T000100	Forceful Browsing

Catalog Update Tools

Detect	Neutralize	Limit	Recover	Classification
N/A	Medium	N/A	N/A	Unclassified
N/A	Medium	N/A	N/A	Unclassified
N/A	Medium	N/A	N/A	Unclassified

Solr Admin (example)

Filter Query: `-!L1:CVE -!L1:CVE*`

Start Row: 0

Maximum Rows Returned: 1000

Fields to Return: id, name, desc

Query Type: standard

Output Type: standard

Debug: enable View: you may need to "view source" in your browser

Debug: explain others Apply original query

Enable Highlighting

Fields to Highlight:

Threat Assessment & Remediation Analysis (TARA)

Approved for Public Release

© 2013 The MITRE Corporation. All rights reserved

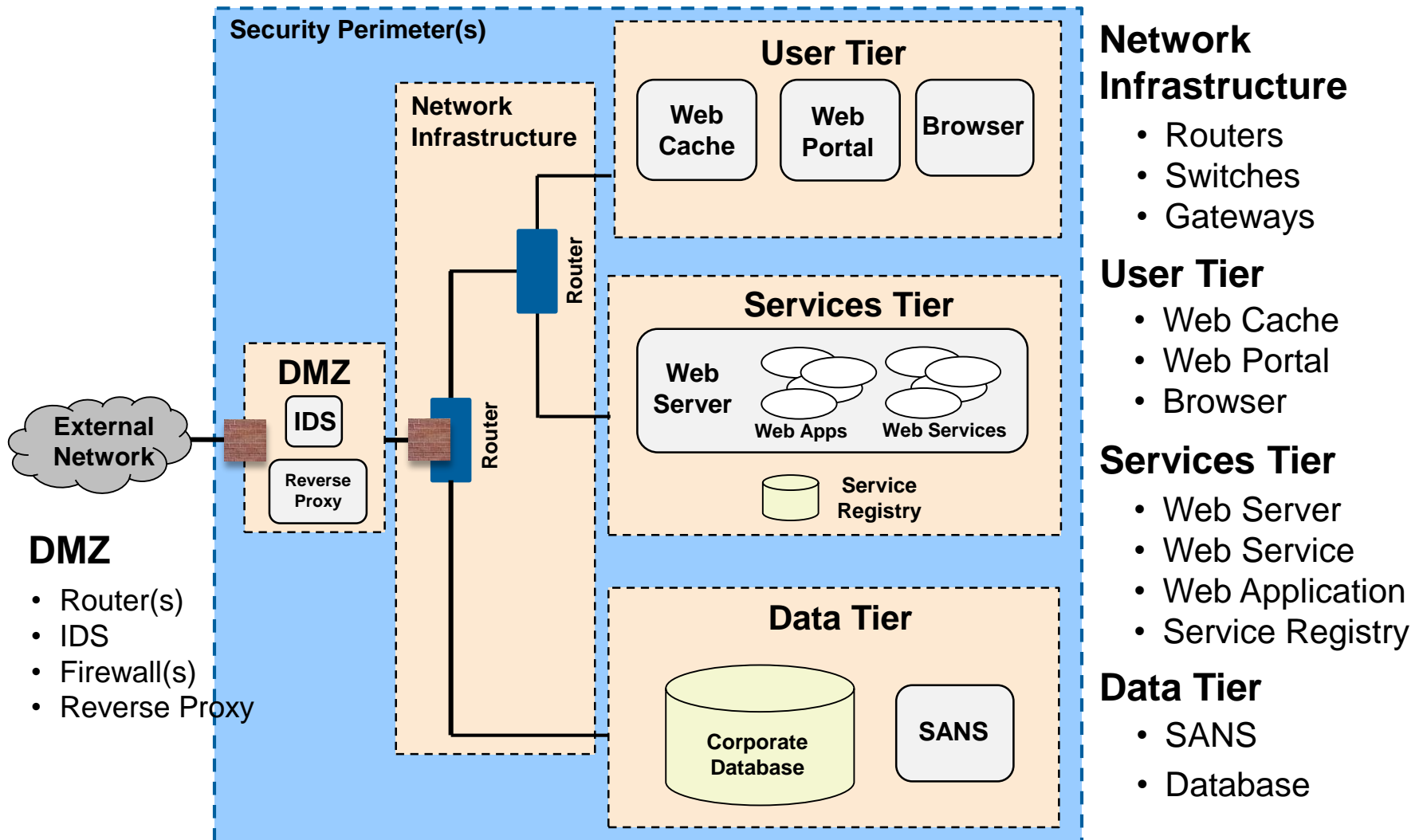
Worked Example

Approved for Public Release

© 2013 The MITRE Corporation. All rights reserved

MITRE

Target(s) of Evaluation



Network Infrastructure

- Routers
- Switches
- Gateways

User Tier

- Web Cache
- Web Portal
- Browser

Services Tier

- Web Server
- Web Service
- Web Application
- Service Registry

Data Tier

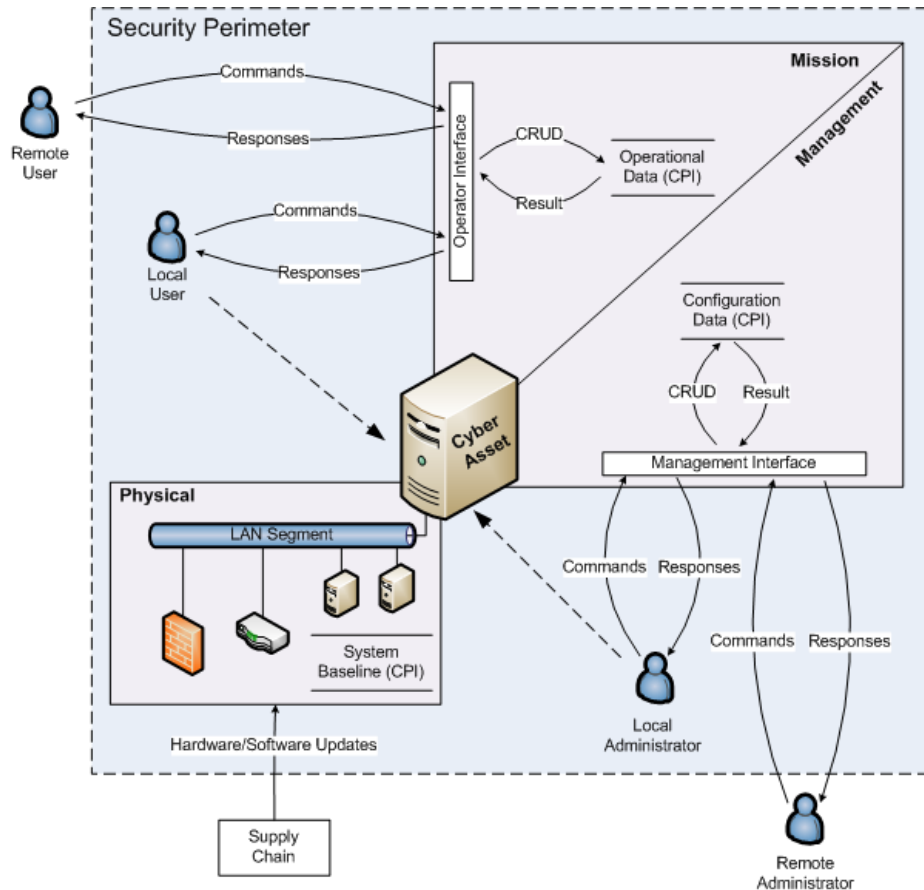
- SANS
- Database

DMZ

- Router(s)
- IDS
- Firewall(s)
- Reverse Proxy

Approved for Public Release

Modeling the Attack Surface



ET modeled in 3 planes

- Mission Plane
- Management Plane
- Physical Plane

System Interfaces(s)

- Standardized functions, CRUD: Create, Read, Update, Delete
- Special purpose algorithms
- May include Critical Program Information (CPI)

Security Perimeter(s)

- Users are local or remote relative to some security perimeter
- Nested perimeters subject to penetration by APT

System data

- Each plane (mission, management, physical) stores and processes data required within that plane
- May include CPI

CRUD – Create, Read, Update, Delete

A model is a simplified representation of a system to facilitate analysis

Filling a Shopping Cart

(with Attack Vectors)

Vector Groups

- Web Server
- Web Service
- Web Application
- Database
- XML
- Web 2.0
-
-

Search by
Vector Group

Search by
Category

Search by
keyword

Catalog Search

Mission Assurance Engineering : Threat Assessment and Remediation Analysis

Home

Records Loaded

Asset Classes

TTPs

Countermeasures

Search for...

Reports

PALMA Reports

Catalog Maintenance

Asset Classes

TTPs

Countermeasures

Admin Functions

Account Management

Catalog Merge Tool

Data Schemas

Spreadsheet Template Converter/Importer

TTP-CM Mapping Tools

TTPs Loaded --- AC IDs in [328]

Apply Filter TTP/CM Mapping

Filter	TTP ID	TTP Name
<input checked="" type="checkbox"/>	T000010	HTTP Request Smuggling
<input checked="" type="checkbox"/>	T000014	Accessing, Intercepting, and Modifying HTTP Cookies
<input type="checkbox"/>	T000016	Simple Script Injection
<input checked="" type="checkbox"/>	T000023	Cross Site Tracing
<input checked="" type="checkbox"/>	T000039	Exploitation of Session Variables, Resource IDs and other Trusted Credentials
<input checked="" type="checkbox"/>	T000056	Web Server/Application Fingerprinting
<input type="checkbox"/>	T000073	HTTP Response Splitting
<input type="checkbox"/>	T000076	HTTP Verb Tampering
<input checked="" type="checkbox"/>	T000078	Flash Parameter Injection
<input type="checkbox"/>	T000081	HTTP Response Smuggling
<input checked="" type="checkbox"/>	T000084	Web Logs Tampering
<input type="checkbox"/>	T000088	Modifying filename extensions to misclassify content
<input checked="" type="checkbox"/>	T000096	Poison Web Service Registry
<input type="checkbox"/>	T000100	Forceful Browsing
<input checked="" type="checkbox"/>	T000101	WSDL Scanning
<input type="checkbox"/>	T000138	Directory traversal
<input checked="" type="checkbox"/>	T000139	Flash Injection

Apply Filter TTP/CM Mapping

Shopping Cart

Mission Assurance Engineering : Threat Assessment and Remediation Analysis

Home

Records Loaded

Asset Classes

TTPs

Countermeasures

Search for...

Reports

PALMA Reports

Catalog Maintenance

Asset Classes

TTPs

Countermeasures

Admin Functions

Account Management

Catalog Merge Tool

Data Schemas

Spreadsheet Template Converter/Importer

TTP-CM Mapping Tools

Search TTPs

Saved Searches: [Select Search] Run Search Modify Search Delete Search

Display Style: Full Minimal

Search Type: Form-Based Standard By AC List

TTP ID: [] TTP Name (use | to separate multiple search terms): []

TTP Categories: Social Engineering, Electronic Warfare, **Asset Class**, Hardware/Firmware, Cyber

Description (use | to separate multiple search terms): software | firmware

Attack Objectives: Disruption, Penetration, Destruction, Exfiltration, Implantation, Recon

Classification Level: []

Origins: Trusted Insider, **Attacker**, Insider

Prerequisites: []

References: []

Save Search: Yes No

Cancel Search



Approved for Public Release

Example Shopping Cart

ID	Attack Vector Name
T000001	BIOS replaced with version that allows unsigned updates
T000008	Unsecured SNMP agent
T000016	Simple Script Injection
T000021	Man in the Middle Attack
T000049	Buffer Overflow
T000105	Embedding Script (XSS) in HTTP Headers

A shopping cart is a collection of attack vectors being evaluated in a TARA assessment.

Attack vectors were picked at random for this example. In an actual TARA assessment, steps to develop a shopping cart include threat modeling, catalog content development, and external research on the system being evaluated and the technologies it incorporates.

Risk Scoring

Factors for assessing TTP Risk				Factor Weight	Attack Vectors					
Factor Range	Low = 1	Medium = 2	High = 3		T000001	T000008	T000016	T000021	T000049	T000105
Locality: How localized are the effects posed by this TTP?	isolated to single unit	external networks potentially impacted	all units globally and associated infrastructure	0.2	1	2	1	2	2	3
Impact: How serious an impact is loss of data confidentiality resulting from successful application of this TTP?	no impact from TTP	limited impact requiring some remediation	Data spills routinely exercised	0.2	2	1	1	1	2	3
Impact: How serious an impact is loss of system availability resulting from successful application of this TTP?	no impact from TTP	limited impact requiring some remediation	Simulated system outages routinely exercised	0.2	1	1	2	2	1	2
Prior Use: Is there evidence that this TTP has been successfully used before?	no evidence of TTP use	confirmed evidence of TTP use	widespread use of TTP reported	0.3	2	3	3	1	2	1
Stealth: How detectable is this TTP when it is applied?	TTP obvious without monitoring	detection likely with routine monitoring	undetectable	0.1	2	2	1	1	1	2
Score				1.0	1.6	1.9	1.8	1.4	1.7	2.1

Risk scoring is an **optional** step in a TARA assessment, which can be performed when a shopping cart includes more attack vectors than can be addressed given time and funding constraints. The spreadsheet above is used to evaluate each attack vector against a set of risk factors. This spreadsheet calculates a risk score for each attack vector as a weighted sum of risk factor values. This scoring approach is intended to rank attack vectors, not to assess absolute risk.

In a TARA assessment, the risk factors, range of values, weightings, and the calculation can all be tailored to the needs of the program or sponsor. The only requirement is that each attack vector in the shopping cart be treated equally in how relative risk is assessed.

Threat Matrix

ID	Attack Vector Name	Risk Score	Evaluation Targets		
			Router	Web Server	Browser
T000105	Cross Site Scripting	2.1			x
T000008	Unsecured SNMP agent	1.9	x		
T000016	Simple Script Injection	1.8	x		x
T000049	Buffer Overflow	1.7	x	x	x
T000001	BIOS replaced with version that allows unsigned updates	1.6	x	x	x
T000021	Man in the Middle Attack	1.4		x	x

*The **Threat [Susceptibility] Matrix** combines shopping cart and risk scoring data across the range of evaluation targets being assessed. This artifact is a primary deliverable and represents the transition from threat susceptibility analysis to risk remediation analysis in the TARA methodology.*

The AV/CM Mapping Table

Countermeasure (CM)			Mitigation Effectiveness (by Attack Vector ID)					
CM ID	Name	Cost Index	T000001	T000008	T000016	T000049	T000021	T000105
C000001	Verify secure BIOS update non-bypassability	Medium	M					
C000002	Verify BIOS image write protection	Low	M					
C000003	Verify recovery process to restore last-known-good BIOS image	Medium	M					
C000005	Institute secure BIOS update capabilities using RTU	High	P					
C000015	Verify BIOS implemented security controls after BIOS image update	Low	P					
C000018	Use checksums to verify the integrity of downloaded BIOS image updates	Low	P					
C000023	Change default SNMP community string values	Very Low		P				
C000024	Restrict SNMP community string value reuse	Low		P				
C000041	Use same character encoding	Medium			P			
C000062	Disable client side scripting	Medium			P			P
C000064	Do not deploy content proxies that mask where data originates from	High			P			
C000065	Sanitize outbound content	High			M			
C000079	Only accept PKI credentials from a trusted certificate authority	Medium					M	
C000081	Use strong mutual authentication	Medium					P	
C000083	Use cryptography that is sufficient strong	Medium					P	
C000090	Validate input fields use of NULL, escape, backslash, meta, and control characters	Medium			M			M
C000101	Verify buffer sizes	Low				M		
C000103	Match buffer size to data input size	Low				M		
C000112	Restrict source of format strings	Low			M			
C000115	Limit user functional roles	Medium						M
C000121	Verify input sources	Medium						P
C000132	Use sandboxing to isolate running software	Medium						M
C000134	Select programming languages that minimize potential software defects	Medium				M		
C000135	Avoid use of dangerous memory functions and operations	Low				M		
C000136	Utilize processor-based protection capabilities	Very Low				M		
C000142	Enforce mutual authentication between communication parties	Medium					P	
C000146	Enable SSL/TLS to protect sensitive web pages	Medium					P	
C000194	Disable hyperlinks in email	Very Low			M			M
C000220	Utilize best practice malware detection approaches	Medium			M			M
C000238	Enforce software quality standards and guidelines that improve software quality	Low				M		
C000247	Ensure trustworthiness of key personnel	Medium					M	

The AVCM mapping table depicts the association of countermeasures to attack vectors in the TARA catalog. Catalog tools provide the means to export mapping table data in spreadsheet form, as depicted. In this example, each mapping characterizes whether a countermeasure has a [P]reventative effect or a [M]itigating effect for each attack vector listed in the threat matrix. A cost index is associated with each countermeasure to reflect the relative cost of ownership for that countermeasure on a linear scale [very low...very high]. These default cost index values can be tailored to reflect truth about the program, e.g., the cost of ownership may be significantly lower if the CM is already implemented as a security measure or security practice in a system that is already fielded.

Approved for Public Release

Countermeasure Scoring

Countermeasure (CM)					Mitigation Effectiveness (by Attack Vector ID)					
CM ID	Name	Cost Index	Utility Score	U/C Ratio	T000105	T000008	T000016	T000049	T000001	T000021
					2.1	1.9	1.8	1.7	1.6	1.4
C000023	Change default SNMP community string values	1	6	6.00		P				
C000062	Disable client side scripting	3	12	4.00	P		P			
C000194	Disable hyperlinks in email	1	4	4.00	M		M			
C000015	Verify BIOS implemented security controls after BIOS image update	2	6	3.00					P	
C000018	Use checksums to verify the integrity of downloaded BIOS image updates	2	6	3.00					P	
C000024	Restrict SNMP community string value reuse	2	6	3.00		P				
C000041	Use same character encoding	3	6	2.00			P			
C000081	Use strong mutual authentication	3	6	2.00						P
C000083	Use cryptography that is sufficient strong	3	6	2.00						P
C000121	Verify input sources	3	6	2.00	P					
C000136	Utilize processor-based protection capabilities	1	2	2.00				M		
C000142	Enforce mutual authentication between communication parties	3	6	2.00						P
C000146	Enable SSL TLS to protect sensitive web pages	3	6	2.00						P
C000005	Institute secure BIOS update capabilities using RTU	4	6	1.50					P	
C000064	Do not deploy content proxies that mask where data originates from	4	6	1.50			P			
C000238	Enforce software quality standards and guidelines that improve software quality	2	3	1.50				M		
C000090	Validate input fields use of NULL, escape, backslash, meta, and control characters	3	4	1.33	M		M			
C000220	Utilize best practice malware detection approaches	3	4	1.33	M		M			
C000002	Verify BIOS image write protection	2	2	1.00					M	
C000101	Verify buffer sizes	2	2	1.00				M		
C000103	Match buffer size to data input size	2	2	1.00				M		
C000112	Restrict source of format strings	2	2	1.00			M			
C000135	Avoid use of dangerous memory functions and operations	2	2	1.00				M		
C000247	Ensure trustworthiness of key personnel	3	3	1.00						M
C000001	Verify secure BIOS update non-bypassability	3	2	0.67					M	
C000003	Verify recovery process to restore last-known-good BIOS image	3	2	0.67					M	
C000079	Only accept PKI credentials from a trusted certificate authority	3	2	0.67						M
C000115	Limit user functional roles	3	2	0.67	M					
C000132	Use sandboxing to isolate running software	3	2	0.67	M					
C000134	Select programming languages that minimize potential software defects	3	2	0.67				M		
C000065	Sanitize outbound content	4	2	0.50			M			

TARA provides a default approach to score countermeasures based on cost benefit analysis. This is an **optional** step used to rank countermeasures prior to their selection. This approach calculates a **Utility/Cost (U/C) ratio** for each countermeasure, based on its cost index and a utility score, which is calculated as the cumulative mitigation value of that countermeasure over the range of attack vectors. In this each example, a score of 6 is assigned to each [P]reventative mapping and a score of 2 is assigned to each [M]itigating mapping. Additionally, the cost index ordinal scale [very low... very high] is remapped to a numeric scale [1...5] in order to compute U/C ratios.

Note that the scores assigned to mappings and the numeric scale used for cost can be tailored to suit the needs of the program. Once a U/C ratio is calculated for each countermeasure, the list is sorted so that countermeasures with higher U/C ratios appear on top and attack vectors are reordered left to right by decreasing risk score.

Approved for Public Release

Countermeasure Selection Strategy

Countermeasure (CM) Selection Strategies

Assurance level: **Low**

For each attack vector

At least 2 CMs total

At least 1 Preventative CM

At least 1 Mitigation CM

Assurance level: **Medium**

For each attack vector

At least 3 CMs total

At least 1 Preventative CM

At least 1 Mitigation CM

Assurance level: **High**

For each attack vector

At least 5 CMs total

At least 2 Preventative CM

At least 1 Mitigation CM

The countermeasure selection strategy establishes constraints on the selection of countermeasures in terms of the minimum number of preventative, mitigating, and total countermeasures required for each attack vector.

In this example, 3 assurance levels are defined: low, medium, high, each requiring progressively more total countermeasures.

This strategy can be tuned to the needs of a particular program or sponsor in terms of the number of countermeasures, the ratio of preventative to mitigating countermeasures etc.

Countermeasure Selection

Countermeasure (CM)		Mitigation Effectiveness (by Attack Vector ID)					
		T000105	T000008	T000016	T000049	T000001	T000021
CM ID	Name	2.1	1.9	1.8	1.7	1.6	1.4
C000023	Change default SNMP community string values		P				
C000062	Disable client side scripting	P		P			
C000194	Disable hyperlinks in email	M		M			
C000015	Verify BIOS implemented security controls after BIOS image update					P	
C000018	Use checksums to verify the integrity of downloaded BIOS image updates					P	
C000024	Restrict SNMP community string value reuse		P				
C000041	Use same character encoding			P			
C000081	Use strong mutual authentication						P
C000083	Use cryptography that is sufficient strong						P
C000121	Verify input sources	P					
C000136	Utilize processor-based protection capabilities				M		
C000142	Enforce mutual authentication between communication parties						P
C000146	Enable SSL/TLS to protect sensitive web pages						P
C000005	Institute secure BIOS update capabilities using RTU					P	
C000064	Do not deploy content proxies that mask where data originates from			P			
C000238	Enforce software quality standards and guidelines that improve software quality				M		
C000090	Validate input fields use of NULL, escape, backslash, meta, and control characters	M		M			
C000220	Utilize best practice malware detection approaches	M		M			
C000002	Verify BIOS image write protection					M	
C000101	Verify buffer sizes				M		
C000103	Match buffer size to data input size				M		
C000112	Restrict source of format strings			M			
C000135	Avoid use of dangerous memory functions and operations				M		
C000247	Ensure trustworthiness of key personnel						M
C000001	Verify secure BIOS update non-bypassability					M	
C000003	Verify recovery process to restore last-known-good BIOS image					M	
C000079	Only accept PKI credentials from a trusted certificate authority						M
C000115	Limit user functional roles	M					
C000132	Use sandboxing to isolate running software	M					
C000134	Select programming languages that minimize potential software defects				M		
C000065	Sanitize outbound content			M			

Countermeasures are selected by applying the countermeasure selection strategy to each attack vector from left to right in the countermeasure scoring table. Countermeasure selection starts at the top of the table where U/C ratios are the highest. Selected countermeasures are highlighted.

In this example, the selection strategy for medium assurance is used. However the strategy cannot be fully satisfied for each attack vector listed, resulting in gaps. Alternative solution sets that meet the strategy constraints can be developed and provide the basis for incorporating sensitivity analysis in the course of a TARA assessment.

Approved for Public Release

Solution Effectiveness

System name: System XYZ				Assurance Level: Medium					
Countermeasure (CM)			Mitigation Effectiveness (by Attack Vector ID)						
ID	Countermeasure Name	Cost Index	T000105	T000008	T000016	T000049	T000001	T000021	
			2.1	1.9	1.8	1.7	1.6	1.4	
C000023	Change default SNMP community string values	1		P					
C000062	Disable client side scripting	3	P		P				
C000194	Disable hyperlinks in email	1	M		M				
C000015	Verify BIOS implemented security controls after BIOS image update	2					P		
C000018	Use checksums to verify the integrity of downloaded BIOS image updates	2					P		
C000024	Restrict SNMP community string value reuse	2		P					
C000081	Use strong mutual authentication	3						P	
C000083	Use cryptography that is sufficient strong	3						P	
C000136	Utilize processor-based protection capabilities	1				M			
C000238	Enforce software quality standards and guidelines that improve software quality	2				M			
C000090	Validate input fields use of NULL, escape, backslash, meta, and control characters	3	M		M				
C000002	Verify BIOS image write protection	2					M		
C000101	Verify buffer sizes	2				M			
C000247	Ensure trustworthiness of key personnel	3						M	
Totals		30	3	2	3	3	3	3	

The solution effectiveness table lists countermeasures that were selected, with each countermeasure detailing the preventative and/or mitigating effects it has on each attack vector assessed. At the bottom of the table a summary is provided to indicate whether the selection strategy was successful for each attack vector, with green indicating success and yellow indicating where a gap exists.

In the example above, T000008 has a gap relative to both the total number of countermeasures applied and the lack of a mitigating countermeasure, while the gap identified for T000049 relates to the absence of a preventative countermeasure. The summary also includes a summary cost index to support comparison with alternative solution sets. This artifact is a primary deliverable and represents completion of the risk remediation phase of a TARA assessment.

Summary

- TARA is an engineering approach that is rigorous and repeatable, provides traceability, identifies gaps, and develops defense-in-depth
- TARA's objective is to influence programs early in the acquisition lifecycle where *the cost of change is less*
- TARA applies model based systems engineering and tradeoff analysis to system security engineering
- TARA maintains and utilizes catalogs of attack vector and countermeasure data that incorporates data from a variety of sources including CAPEC, CWE, and CVE
- The TARA approach is flexible and can be tailored to meet the needs of MITRE sponsors and programs
- TARA has been applied to several Army, Navy, Air Force, and DoD acquisition programs

For More Information

Online information and resources

<http://www.mitre.org/publications/technical-papers/threat-assessment--remediation-analysis-tara>

<http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/systems-engineering-for-mission-assurance/cyber-risk-remediation-analysis>

<http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/systems-engineering-for-mission-assurance/cyber-threat-susceptibility-assessment>

To schedule a demo, consultation, or for general inquiries

Jackson Wynn
(781) 271-3419
jwynn@mitre.org

