

NICF - Linux System Administration (SF)

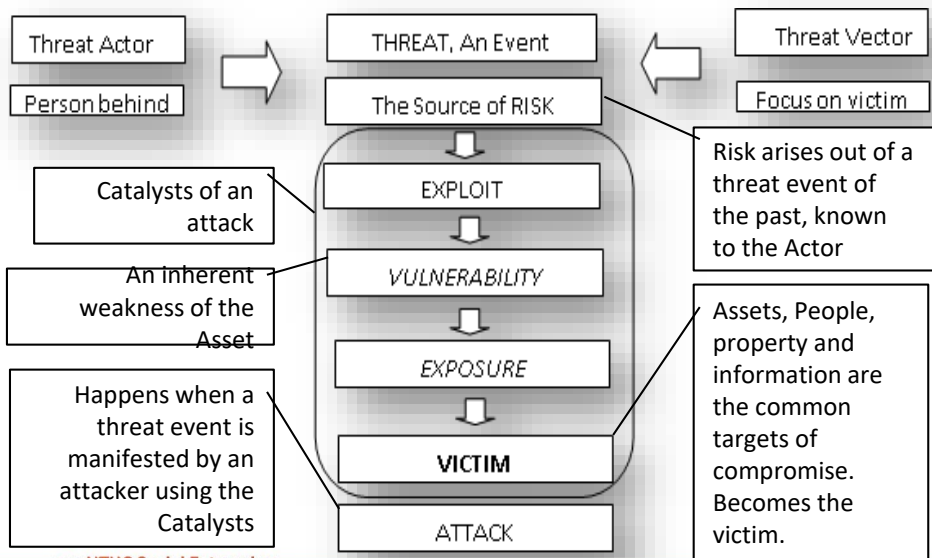
20 Slides Supplement to the trainer's
PP file form Linux Foundation

SF Competency Standard ICT-DES-3004-1.1



Section 23: Alignment with SF TSC

Asset-Threat-Risk-Vulnerability



Asset-Threat-Risk-Vulnerability

Asset – People, property, and information.

People may include employees and customers along with other invited persons such as contractors or guests.

Property assets consist of both tangible and intangible items that can be assigned a value. Intangible assets include reputation and proprietary information.

Information may include databases, software code, critical company records, and many other intangible items.

An asset is what we're trying to protect.

Asset-Threat-Risk-Vulnerability

Threat – Anything that can exploit a vulnerability, intentionally or accidentally, and obtain, damage, or destroy an asset.

A threat is what we're trying to protect against.

Vulnerability – Weaknesses or gaps in a security program that can be exploited by threats to gain unauthorized access to an *asset*.

A vulnerability is a weakness or gap in our protection efforts.

Risk – The potential for loss, damage or destruction of an asset as a result of a threat exploiting a *vulnerability*.

Risk is the intersection of assets, threats, and vulnerabilities.

Asset-Threat-Risk-Vulnerability

It important to understand the difference between these terms

A formula used to determine risk is.... **A + T + V = R**

Risk is a function of threats exploiting vulnerabilities to obtain, damage or destroy assets. Thus, threats (actual, conceptual, or inherent) may exist, but if there are no vulnerabilities then there is little/no risk.

Similarly, you can have a vulnerability, but if you have no threat, then you have little/no risk.

Accurately assessing threats and identifying vulnerabilities is critical to understanding the risk to assets. Understanding the difference between threats, vulnerabilities, and risk is the first step.

Asset-Threat-Risk-Vulnerability

The **threat** of a hurricane is outside of one's control. However, knowing that a hurricane could potentially hit can help business owners assess weak points and come up with an action plan to minimize the impact. In this scenario, a **vulnerability** would be not having a data recovery plan in place in the event that your physical assets are damaged as a result of the hurricane's winds or heavy rains. The **risk** to your business would be the loss of information or a disruption in business as a result of not addressing your vulnerabilities.

Identification of Threat-Risk-Vulnerability

Accurately identifying and assessing threats and vulnerabilities is critical to understand the risk to assets.

What Is Threat Modelling?

Threat modelling is a proactive approach to identify entry points to enumerate threats and building security to prevent security breaches in applications and computer systems.

Threat models allow organizations to identify, predict, and define internal and external security threats from the attacker's point of view, and protect their valuable assets by building a defence strategy that prioritizes resources and prepares the appropriate response. Threat models provide a comprehensive view of the organization's full attack surface, helping CISOs and other security team members to make decisions on how to prioritize security actions.

Threat Modelling : Tools and techniques

As *threat modelling* methodologies evolve, security professionals are recognizing the importance of choosing the right threat modelling methodology for an organization's specific challenges and the rise of new threats.

From a theoretical perspective, each threat modelling methodology provides security teams and organizations with the means to identify potential threats and may be seen on equal footing. However, on a practical level, threat modelling methodologies vary in quality, consistency, and value received for the resources invested.

Threat Modelling : Tools and techniques

What Users Want in a Threat Modeling Methodology					
	OCTAVE	Trike	P.A.S.T.A.	Microsoft	VAST
Implement application security at design time	✓	✓	✓	✓	✓
Identify relevant mitigating controls	✓	✓	✓	✓	✓
Directly contributes to risk management	✓	✓	✓		✓
Prioritize threat mitigation efforts	✓	✓	✓		✓
Encourage collaboration among all stakeholders	✓	✓			✓
Outputs for stakeholders across the organization	✓				✓
Consistent repeatability		✓			✓
Automation of threat modeling process		✓			✓
Integrates into an Agile DevOps Environment					✓
Ability to scale across thousands of threat models					✓

OCTAVE	The Operationally Critical Threat, Asset, and Vulnerability Evaluation methodology[1] were one of the first created specifically for cybersecurity threat modelling
STRIDE	A documented and well known owing to Microsoft's significant influence in the software industry and their offering of Microsoft TMT.
P.A.S.T.A.	The PASTA threat modelling methodology combines an attacker-centric perspective on potential threats with risk and impact analysis.
VAST	The Visual, Agile, and Simple Threat modelling (VAST) methodology was conceived after reviewing the shortcomings and implementation challenges inherent in the other modelling methodologies
Trike	Trike threat modelling is an open source threat modelling methodology focused on satisfying the security auditing process from a cyber risk management perspective

Vulnerability Assessment Steps

- | | |
|---|--|
| 1 | Identify and realize the approach of your company or industry like how it is structured and managed. |
| 2 | Trace the data, systems, and applications that are exercised throughout the practice of the business |
| 3 | Examine the unobserved data sources capable of allowing simple entry to the protected information |
| 4 | Classify both the virtual and physical servers that run the essential business applications |
| 5 | Track all the existing security measures which are already implemented |
| 6 | Inspect the network for any vulnerability |

Threat Modelling : Tools and techniques

Adversaries	Attributes	Goals/results	Methods
Insider	• Employee • Contractor • Temporaries • Former Employees • Student • Vendor	• Revenge • Retaliation • Money • Ideology • Sabotage	• Destruction • Spoofing • Disruption of service • Trap doors • Virus • Trojans
Hacker	• Access to sophisticated hardware and software • Generally non-violent • Technical competence	• Distinction-Celebrity • Vandalism • Revenge • Retaliation	• Destruction • Spoofing • Denial of Service • Social engineering
Criminals	• Sometimes violent • Access to sophisticated hardware and software	• Protect of Operation • Vandalism Arson • Blackmail • Financial gain	• Kidnapping • Destruction • Spoofing • Disruption of service • Social engineering

Adversaries	Attributes	Goals/results	Methods
Corporations	• Attempts to collect protected information • Has support: ○ Technical, ○ Analytical ○ Financial	• Corporate Espionage • Money • Financial gain	• Social engineering • Spoofing • Trap doors
Government Agencies	• Trained in espionage • Possesses equipment • Has support: ○ Technical, ○ Analytical, ○ Financial.	• Disruption of service • Destruction	• Destruction • Spoofing • Disruption of service • Kidnapping • Social engineering
Terrorist	• Technical competence • Access to sophisticated hardware and software • Violent • Politically motivated	• Destruction of capability • Political statements • Sabotage • Espionage	• Destruction • Spoofing • Disruption of service • Kidnapping • Social engineering
Disasters	• Natural events	• Disruption of service • Destruction	• Fire, Earthquake, • Lighting • Storms • Utility break downs

Risk Identification tools and techniques

Interview with SME's

Brainstorming

Delphi Technique

Checklist

Cause and Effect Diagram (Fishbone Diagram)

Business Impact Analysis

SWOT (Strengths Weaknesses Opportunities Threats)

Crawford Slip

Risk Assessment & Cost benefit analysis

Annualized Loss Expectancy (ALE) = Single Loss Expectancy x Annualized Rate of Occurrence.

Single Loss Expectancy (SLE) = Asset Value x Exposure factor

Annualized Rate of Occurrence (ARO) = Estimated frequency a threat will occur within a year and is characterized on an annual basis.

Exposure Factor (EF) = Percentage of asset loss caused by identified threat; ranges from 0 to 100

Safeguard cost/benefit analysis =

(ALE before implementing safeguard) – (ALE after implementing safeguard) – (annual cost of safeguard)

Asset value (AV) = \$ 100,000.00, Exposure factor (EF) = 25% ARO = 0.04 then ALE = \$1000.00

Asset value (AV) = \$ 100,000.00, Exposure factor (EF) = 25% ARO = 0.01 then ALE = \$400.00

Annual cost of safeguard lesser than, \$600.00

IS Standards & Frameworks

ISO/IEC 27001/2	ISO 2700X standard gives guidelines for organizational IS standards and IS management practices including implementation and management of controls, taking into consideration the organization's IS risk environment(s)
NIST 80053 /CSF	A high-level document focuses on how to access and prioritize security functions, and references existing documents
CIS 20	The Center for Internet Security maintains a standard of 20 controls, originally developed by SANS.
ISACA COBIT 5	The Information Systems Audit and Control Association offers the COBIT framework for information security.
ISF	The Information Security Forum's Standard of Good Practice provides controls and guidance on current and emerging security topics
DISA	The Defense Information Security Agency provides security standards including Control Correlation Identifiers (CCIs), Security Requirements Guides (SRGs), Security Technical Implementation Guides (STIGs), and mobile code
ITIL	ITIL is directed to IT Service Management. Aligned with the ISO 20000 series, it is published as a series of five core volumes, each covering a different ITSM lifecycle stage
PCI-DSS	A joint venture by the major credit card companies, the Payment Card Industry security council's Data Security Standard is a set of policies and procedures intended to improve the security of card transactions

Linux Security Tips and Checklist

Encrypt Data Communication for Linux Server

Keep Linux Kernel and Software Up to Date

Minimize Software to Minimize Vulnerability in Linux

Avoid Using FTP, Telnet, And Rlogin / RSH Services on Linux

Use Linux Security Extensions

One Network Service Per System or VM Instance

Linux Security Tips and Checklist

Encrypt Data Communication for Linux Server

Keep Linux Kernel and Software Up to Date

Minimize Software to Minimize Vulnerability in Linux

Avoid Using FTP, Telnet, And Rlogin / RSH Services on Linux

Use Linux Security Extensions

One Network Service Per System or VM Instance

Linux Tool AutoYaST2

- It is available with recent SUSE products Linux 8.0.
- The steps following the proposal are fully automated and the user is ONLY prompted to configure hardware and network services
- Control file uses XML to describe the system installation and configuration
- RELAX NG schema does specifies a pattern for the structure and content of an XML document.
- If necessary YaST modules are NOT available in the system, important configuration steps will be skipped.
- In openSUSE 11.1 allows configuration of the proposal screen with the `<proposals config:type="list">` option in the profile.

Linux Authentication: PAM Tool

Linux uses PAM (pluggable authentication modules) in the authentication process as a layer that mediates between user and application. PAM modules are available on a systemwide basis, so they can be requested by any application. This chapter describes how the modular authentication mechanism works and how it is configured.

System administrators and programmers often want to restrict access to certain parts of the system or to limit the use of certain functions of an application. Without PAM, applications must be adapted every time a new authentication mechanism, such as LDAP, Samba, or Kerberos, is introduced. This process, however, is rather time-consuming and error-prone. One way to avoid these drawbacks is to separate applications from the authentication mechanism and delegate authentication to centrally managed modules. Whenever a newly required authentication scheme is needed, it is sufficient to adapt or write a suitable PAM module for use by the program in question.

Network Usage of Processes: nethogs

PID	USER	PROGRAM	DEV	SENT	RECEIVED
27145	root	zypper	eth0	5.719	391.749 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30015		0.102	2.326 KB/sec
26635	tux	/usr/lib64/firefox/firefox	eth0	0.026	0.026 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30045		0.000	0.021 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30045		0.000	0.018 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30015		0.000	0.018 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30045		0.000	0.017 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30045		0.000	0.017 KB/sec
?	root	..0:113:80c0:8080:10:160:0:100:30045		0.069	0.000 KB/sec
?	root	unknown TCP		0.000	0.000 KB/sec
TOTAL				5.916	394.192KB/sec

M: cycle between display modes (kb/s, kb, b, mb)

R: sort by RECEIVED

S: sort by SENT

Q: quit

The End