

**Beware of Geeks bearing gifts!** 

### Larry England and John Krautheim 04 May 2022



Copyright © 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.

### **Table of Contents**

- Who are we?
  - Intro & Goals
    - Approach a simple 4-step tango Dancing anyone?
      - Examine Log4j vulnerability



## Who are the presenters? Who are these guys anyway?

#### John Krautheim, PhD, CISSP-ISSEP

Broadcom Software Engineer - Pittsburgh 20 years experience in Computer Security Enjoys cycling and camping

Is an audiophile on the cheap



#### Larry England

Broadcom Software Engineer with experience across many technologies - sunny (rainy?) California

Enjoys hiking, trail running (ultras), crashing bikes, xcountry skiing, photography, music

Very amatuer piano player

Participant in the witness protection program





# Once upon a time, there was Log4j...





### Where were you when Log4j hit you? ...

On or about Dec 10th, 2021, while at work, I **awoke to the question** by one of our "Security Champions" to hear

"Who is using Log4j()???"... and

The Slack Channel is going nuts

See <u>CVE2021-44228</u> CVSS score 10!! Outline of steps to exploit see <u>this.</u>

Estimated number of attempted attacks was 10 million to exploit this zero-day vulnerability!

# Could this have been prevented? How might this have been prevented? we'll look at this question later ....

5 | Copyright © 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.



Intro







Assuming every system is penetrable (if turned on) and you are either a general user or external user

ask yourself these questions:

How would you breach these systems? Where are the vulnerabilities? What are the risks for our users?



7 | Copyright © 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.

# What is a Threat?

A potential event that has "unwelcome/unintended consequences".

Or an individual or org from which an attack can originate.

Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service. Also, the potential for a threat-source to successfully exploit a particular information system vulnerability.

- NIST SP 1800-15B

#### The big questions to ask:

- What are we working on?
- What could go wrong?
  - What is the probability an identified threat will happen?
  - What would the consequences/costs be if it does happens?
- What are we going to do about it?
- Did we do a good job?





# What is Threat Modeling?



Set of approaches and techniques to identify risks affecting a system based on

- how it is architected
- how is it coded
- how it is supposed to work

Through **software design analysis**, threat modeling identifies weaknesses by juxtaposing various design views against threat agents. This includes

- Security controls and boundaries
  - ex: not mixing authorized and unauthorized code in an address space
- Weak or ineffective encryption
  - ex: use of DES-56 encryption or using your own encryption (even modified standard encryption)
- Potential vulnerabilities
- Risk Scenarios What could go wrong go wrong go wrong go wrong go wrong go wrong ?



### Think like a hacker. Break the rules. If not us, someone else will.

- **Goal**: get the system(s) to behave in an unintended manner ... somehow!
- Assume you can obtain a general user credentials (even on z)
  - How can you impersonate another (elevated) id?
  - Are credentials passed in the clear?
  - Can the various credentials be compromised/spoofed?
  - What are the credentials for datastores? Can datastores be compromised? What could I find in datastore?
- What are the opportunities for input by users?
  - How do you ensure your caller is whom you expect/allow?
  - What does the protocol allow? How is it protected from abuse/unauthorized actions?
- Can a man-in-the-middle attack be conducted due to the number of hops?
  - Does any information get 'leaked' that could be used in combination with other info to form an attack (like leaking userids, (potentially sensitive) applications, etc.)?
- Policy enforcement where defined? how to circumvent? how enforced?
  - How is an aberrant/malicious application prevented from being scheduled on an agent(s)?
  - Is it possible to install a Trojan horse into a system?
- Are there vulnerabilities open on z/OS due to this structure?





## 4 easy steps to Security Sobriety (threat modeling)

- **1**. Decompose the application / product
- 2. Find the threats think like a hacker
- 3. Rank the threats
- 4. Determine steps to take reduce the threats



### **Step 1 - Decompose the Application / Product**



# **Steps - How to approach this task?**

Model the application to understand how a system works - Asking "what could go wrong?" **looking at "4+1 architecture"** 

- 1) Identify trust boundaries/zones
- 2) Identify actors (internal and external)
- 3) Add data flows
- 4) Identify potential entry points
- 5) Identify risks / assess impacts
- 6) Authentication flows
- 7) Dependencies
- 8) Entry & exit points
- 9) Assets (ex: databases, files)

Things to think about - can you create a doomsday scenario - what's the worst that could happen?



# 4+1 Architectural model - provides different perspectives of the arch

**Logical View** - functionality of the systems for the end-users

**Process View** - The system processes and how they communicate / trust boundaries

**Development View** - implementation view

Physical View - Deployment view / topology

Scenarios/use cases





## **Step 2 - Find the Threats**



### **Security properties to consider**

property	description
Confidentiality	Data is only available to the people intended to access it.
Integrity	Data and system resources are only changed in appropriate ways by appropriate people.
Availability	Systems are ready when needed and perform acceptably.
Authentication	The identity of users is established (or you're willing to accept anonymous users).
Authorization	Users are explicitly allowed or denied access to resources.
Non-repudiation	Users can't perform an action and later deny performing it



### Use a well-known, structured approach - STRIDE



#### **S**poofing

Tampering

Repudiation

Information disclosure

**D**enial of Service

Elevation of privilege



### **<u>STRIDE</u>** = **S**poofing, **T**ampering, **R**epudiation, **I**nfo disclosure, **D**oS, **E**levation of Privilege

#### A structure to consider when performing a threat assessment

Spoofing	accessing and use of another user's credentials	authentication
Tampering	maliciously change or modify persistent data, such as a database	integrity
Repudiation	performing prohibited operations in a system that lacks the ability to trace the operations	non-repudiation
Info Disclosure	intending to read a file that one was not granted access to, or to read data in transit.	confidentiality
Denial of Service	attempting to deny access to valid users, such as by making a web server temporarily unavailable or unusable	availability
Elevation of Privilege	intending to gain privileged access to resources in order to gain unauthorized access to information or to compromise a system	authorization





### Why do we need all of the '4 models/flows'?

Element	Spoofing	Tampering	Repudiation	Information Disclosure	Denial of Service	Elevation of Privilege
Data Flows		Х		Х	Х	
Data Stores		Х		Х	Х	
Processes	Х	Х	Х	Х	Х	Х
Interactors	Х		Х			
Trust Zones						Х



## **Step 3 - Rank the Threats**



### **DREAD** - how to assess impact of a vulnerability

- Damage: How big would the damage be if the attack succeeded?
- **R**eproducibility: How easy is it to reproduce an attack?
- Exploitability: How much time, effort, and expertise is needed to exploit the threat?
- Affected Users: If a threat were exploited, what percentage of users would be affected?
- **D**iscoverability: How easy is it for an attacker to discover this threat?







### MITRE ATT&CK®

### What is <u>MITRE ATT&CK<sup>®</sup></u> (Adversarial Tactics, Techniques and Common Knowledge)?

It's a knowledge base of **tactics** and **techniques** designed for threat hunters, defenders and red teams to help classify attacks, <u>identify attack attribution and objectives</u>, and <u>assess an organization's risk</u>.

- Constantly evolving based upon new attack vectors discovered
- Very detailed (eye chart if shown in a slide!) find it here -> <u>https://attack.mitre.org/</u>
- It is presented in a tabular form
  - columns that represent the tactics (or desired outcomes) used during the life of an attack
  - rows that represent of techniques that are utilized to achieve their tactical goals.
  - over 400 attack patterns identified

#### Why even look at MITRE ATT&CK?

- Adopt an attacker's perspective
- The benefit of the ATT&CK framework is that organizations can gain an understanding of how adversaries operate, the steps they might plan to take to gain initial access, discover, move laterally, and exfiltrate data.

#### how to use the MITRE ATT&CK framework?

- Good career :-)
- See this doc from the US Center for Cybersecurity and Infrastructure Security Agency (CISA)





ATT&CK v13 has been released! Check out the blog post or release notes for more information.

MITRE ATT&CK<sup>®</sup> is a globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.

With the creation of ATT&CK, MITRE is fulfilling its mission to solve problems for a safer world – by bringing communities together to develop more effective cybersecurity. ATT&CK is open and available to any person or organization for use at no charge.

 ATTR&CK®

 Getting Started
 Take a Tour

 Contribute
 Biog C

 FAQ
 Random Page

Tweets by MITREattack

#### ATT&CK Matrix for Enterprise

layout: side - show sub-techniques hide sub-techniques

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
10 techniques	8 techniques	9 techniques	14 tecnniques	19 tecnniques	13 techniques	42 techniques	1 / techniques	31 techniques	9 techniques	17 techniques	16 techniques	9 techniques	13 techniques
Active Scanning (3)	Acquire Access	Drive-by Compromise	Cloud Administration Command	Account Manipulation (5)	Abuse Elevation Control	Abuse Elevation Control	Adversary-in-the- Middle (a)	Account Discovery (4)	Exploitation of Remote Services	Adversary-in-the- Middle (n)	Application Layer	Automated Exfiltration (1)	Account Access Removal
Gather Victim Host Information (4)	Acquire Infrastructure (8)	Exploit Public-Facing Application	Command and Scripting	BITS Jobs	Access Token	Access Token Manipulation (5)	Brute Force (4)	Application Window Discovery	Internal Spearphishing	Archive Collected	, Communication	Data Transfer Size	Data Destruction
	Compromise		Interpreter (9)		Manipulation (5)	many a bar		Browser Information Discovery		Data (3)	Through Removable	Limits	Data Encrypted for Impact
Information (3)	Accounts (3)	Services	Container Administration	Execution (14)	Boot or Logon Autostart	BITS JODS	Password Stores (5)	Cloud Infrastructure Discovery	Lateral fool fransfer	Audio Capture	Media	Exfiltration Over	Data Manipulation (3)
Gather Victim Network	Infrastructure (7)	Hardware Additions	Command	Boot or Logon	Execution (14)	Build Image on Host	Exploitation for	Cloud Service Dashboard	Session Hijacking (2)	Automated Collection	Data Encoding (2)	Protocol (3)	Defacement (2)
Information (6)	Develop Capabilities (a)	Phishing (n)	Deploy Container	Initialization Scripts (5)	Boot or Logon Initialization Scripts (m)	Debugger Evasion	Credential Access	Cloud Service Discovery	Remote Services (7)	Browser Session	Data Obfuscation (3)	Exfiltration Over C2	Disk Wipe (2)
Gather Victim Org			Exploitation for Client	Browser Extensions		Deobfuscate/Decode Files or	Forced Authentication			Hijacking	Dynamic Resolution (3)	Channel	
Information (4)	Establish Accounts (3)	Replication Inrough Removable Media	Execution	Compromise Client	System Process (a)	Information	Forge Web	Cloud Storage Object Discovery	Removable Media	Clipboard Data	Encrypted Channel (2)	Exfiltration Over Other	Service (n)
Phishing for Information (3)	Obtain Capabilities (6)		Inter-Process	Software Binary	(4)	Deploy Container	Credentials (2)	Container and Resource				Network Medium (1)	
Search Closed Sources (2)	Stage Canabilities (6)	Supply Chain Compromise (2)	Communication (3)	Create Account (a)	Domain Policy Modification (2)	Direct Volume Access	Input Capture (a)	Discovery	Software Deployment Tools	Data from Cloud Storage	Fallback Channels	Exfiltration Over	Firmware Corruption
bearen biosea boardes (2)	ouge opparinges (0)	Compromise (3)	Native API	orcore Hocoant (3)	modification (2)	Direct Volume Access	import oupture (4)	Debugger Evasion		otoruge	Ingress Tool Transfer	Physical Medium (1)	Inhibit System Recovery
Search Open Technical Databases (n)		Trusted Relationship	Scheduled Task/Job m	Create or Modify System Process (i)	Escape to Host	Domain Policy Modification (2)	Modify Authentication	Device Driver Discovery	Taint Shared Content	Data from Configuration	Multi-Stane Channels	Exfiltration Over Web	Network Denial of
5000000(3)		Valid Accounts (4)		oforcent rococo (a)	Event Triggered	Execution Guardrails (1)	1 100000 (8)	o on our of or of our of o	Use Alternate	Repository (2)	more orage origination	Service (3)	Service (2)
Search Open Websites/Domains and			Serverless Execution	Event Triggered	Execution (16)	Exploitation for Defense	Multi-Factor	Domain Trust Discovery	Authentication Material (a)	Data from Information	Non-Application Layer	Scheduled Transfer	Resource Hilsching
reported portuins (3)			Shared Modules	Excountry (10)	Exploitation for Privilege	Evasion	Interception	File and Directory Discovery	(motorioi (4)	Repositories (3)		Generatica Transfer	Tresource Higheriting
Search Victim-Owned			Software Deployment	External Remote	Escalation	File and Directory Permissions	Multi-Eactor	Group Policy Discovery		Data from Local	Non-Standard Port	Transfer Data to	Service Stop
Websites			Tools	Gervicea	Hijack Execution	Modification (2)	Authentication	croup rolley blacorery		System	Protocol Tunneling	cloud Account	System Shutdown/Reboot
			Quatern Convicer	Hijack Execution	Flow (12)	Mide Artifacte	Request Generation	Network Service Discovery		Data from Natwork	Drown		
			System Services (2)	(12)	Process Injection (12)	Hide Artifacts (18)	Network Sniffing	Network Share Discovery		Shared Drive	rioxy (a)		

BROADCOM<sup>®</sup>

### **MITRE CAPEC™**

#### What is <u>MITRE CAPEC</u><sup>™</sup> (Common Attack Pattern Enumerations and Classifications) ?

A comprehensive dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. It can be used by analysts, developers, testers, and educators to advance community understanding and enhance defenses.

- CAPEC is focused on application security and describes the common attributes and techniques employed by adversaries to exploit known weaknesses in cyber-enabled capabilities. (e.g., SQL Injection, XSS, Session Fixation, Clickjacking)
  - Focus on application security
  - Enumerates exploits against vulnerable systems
  - Includes social engineering / supply chain
  - Associated with Common Weakness Enumeration (CWE)
- Domains of Attack
  - Software
  - Hardware
  - Communications
  - Supply Chain
  - Social Engineering
  - Physical Security
- Details are here -> <u>https://capec.mitre.org/</u>



🚳 🛛 🕅 CAPEC - CAPE	C-3000: Domains × +					~				×
$\rightarrow$ C	O A https://capec.mitre.org/data	/definitions/300		∎ੇ <b>ਟ</b>	3				பி	≡
	Common Attack Pattern I Community Resource for Identi C-3000: Domains of Attack (Version 3.9	Enumerat fying and Ur	ion and Classification derstanding Attacks			ID	Lookup	Nev CAP Start	w to EC? Here!	^
	Home	About	CAPEC List Community	News Search						
APEC VIEW	: Domains of Attack	r.								
View ID: 3000 Structure: Graph										
					Downloa	ads: <u>Boo</u>	klet   <u>C</u>	<u>5V   X</u>	IML	
<sup>r</sup> Objective										
This view organizes	attack patterns hierarchically ba	sed on the	attack domain.						_	
Relationships										
The following graph group patterns that methodology or tec	shows the tree-like relationship share a common characteristic. hnique. Below these are standar	s between a Within cate d and detail	ttack patterns that exist at different jories, meta level attack patterns a ed level patterns that are focused or	t levels of abstraction. At the hi re used to present a decidedly n a specific methodology or teo	ighest level, ca abstract charae chnique used.	ategori cteriza	es exis tion of	a a		
						SI	now Deta	ils: 🗌	J.	
		I	xpand All   Collapse All   Filter View							
	f Attack (513) (515) ations - (512) in - (432) neering - (403) curity - (514)						BACK TO	o top		
V Notes										
Other When this view is a patterns) can be a View Metrics	fully expanded, only the immedi ccessed by opening up the meta	ate children CAPEC entr	(meta patterns) of the top level cat ies. This is a known issue and will b	egories will be visible. Lower le le corrected in a future release	evel children (s	tandar	d and	detail	led	
	CAPECs in this view		Total CAPECs						ł.	
Attack Patterns Categories	559	out of	259 21							
Views	0	out of	13							
Total	565	out of	593							
Content History										
										~
adcom All Rights Res	served The term "Broadcom" refers	o Broadcom	nc and/or its subsidiaries							Л

25 | Copyright © 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.

# Using ATT&CK and CAPEC

- Use ATT&CK for:
  - Comparing computer network defense capabilities
  - Defending against the Advanced Persistent Threat
  - Hunting for new threats
  - Enhancing threat intelligence
  - Adversary emulation exercises

- Use CAPEC for:
  - Application threat modeling
  - Developer training and education
  - Penetration testing



# **Kill Chain**



https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/photo/cyber/ THE-CYBER-KILL-CHAIN-body.png.pc-adaptive.full.medium.png



## **Step 4 - Actions to Take**



### Actions to take

• Open a work item to track the work (this may be in a separate bug repo)

• Report the findings so we can learn from the experience/exercise - ie provide feedback to the engineering team to learn an elevate their expertise

## **Revisit the infamous Log4j**





# Can we use Log4j as an example??

What if threat model was performed against Log4j ... would the huge hole been discovered?

See <u>Threat Modeling as a way of Thinking about Design Flaws</u>





# Is Log4j in your system?



https://www.ibm.com/common/ssi/GIF/ALET/AIM00010.GIF



## **CICS using Java Pipeline**



https://www.ibm.com/docs/en/cics-ts/5.3?topic=caspjr-cics-as-service-provider-json-requests-using-cics-java-pipelines



# **CICS Configuration File**

Project setup:

Log4jBundle project (OSGI bundle)

contains MANIFEST.MF

contains log4j.jar

contains log4j.properties

HelloWorldBundle project (OSGI bundle) contains MANIFEST.MF

HelloWorldCICSBundle project (CICS bundle)

contains cics.xml

contains Log4jBundle

contains HelloWorldBundle



### 4-in-1 Model for Java application with Log4j





# What could possibly go wrong?

- What if the log messages include JNDI lookups?
- What if we use JNDI to connect to an external LDAP server?
- What if the LDAP server is controlled by a third actor?
- What if the LDAP server responds with directory information pointing to another external service under an attacker's control?
- What if JNDI can be used to cause remote code execution via deserialization?



### This could happen



https://cyberint.com/blog/research/log4j-incident-update/



## Summary

- Think like a hacker! Nothing takes the place of thinking!
- Adopt Threat Modeling
- ToDo: Implement Threat Modeling into your culture think like a hacker!



Alvaro Muñoz @pwntester

If developers dont know that untrustred data should not be passed to a JNDI lookup op then WE (the security community) have failed them. Its not THEIR fault







Copyright © 2023 Broadcom. All Rights Reserved. The term "Broadcom" refers to Broadcom Inc. and/or its subsidiaries.

# **Bibliography**

- → Threat Modeling Manifesto
- https://www.iriusrisk.com/resources-blog/threat-modeling-as-a-way-of-thinkingabout-design-flaws-log4j-case
- Threats What Every Engineer Should Learn from Star Wars
- → <u>https://attack.MITRE.org/</u>
- → <u>https://capec.MITRE.org/</u>
- https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-th reats
- → <a href="https://satoricyber.com/glossary/threat-modeling-with-microsoft-dread/">https://satoricyber.com/glossary/threat-modeling-with-microsoft-dread/</a>
- https://shapingsoftware.com/4-1-view-model-of-software-architecture/
- → <u>https://owasp.org/www-community/Threat\_Modeling\_Process</u>

