[I03] decide what you need to do

[I03_t01] risk assessment

wim mees

introduction

learning objectives

- be able to perform threat modeling
- be able to evaluate risks
- understand how risks can be mitigated

threat modeling

data flow diagram

process

- rounded rectangle or circle
- any running code (e.g. written in C, php, . . .)

Figure 1: process

data flow diagram

data store

- two horizontal parallel lines with a label between them
- anything that stores data (e.g. files, databases, Windows registry, shared memory segments, . . .)

Figure 2: data store

data flow diagram

data flow

- arrow
- any exchange between processes or between processes and data stores (e.g. HTTP connections, RPC, ...)

Figure 3: data flow

data flow diagram

external entities

- rectangle with sharp corners
- people or code outside our control (e.g. human user, external web service, . . .)

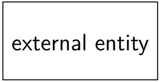


Figure 4: data flow

data flow diagram

example

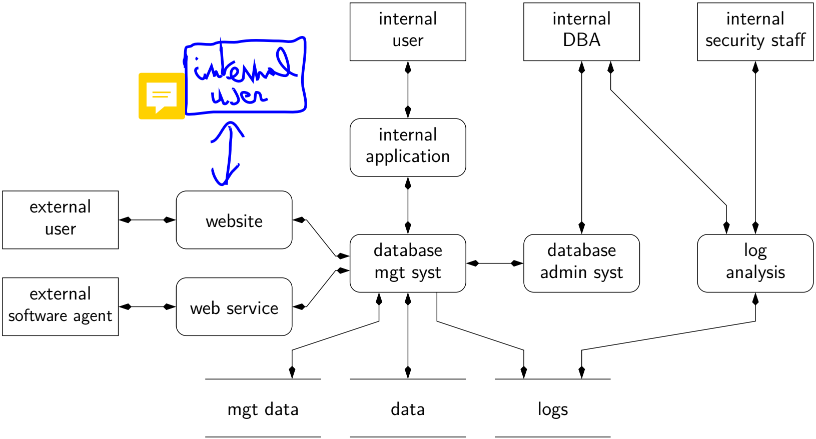


Figure 5: example

trust boundaries

- boundaries are drawn to show who controls what
- these are called *"trust boundaries"*
- threats that cross trust boundaries are typically important ones
- a trust boundary and an *"attack surface"* are very similar

Figure 6: trust boundary

trust boundaries

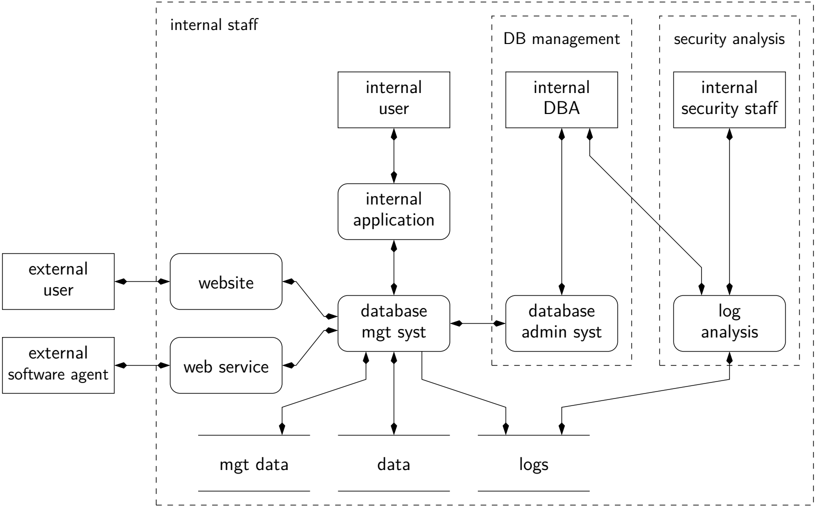


Figure 7: example

STRIDE

STRIDE threats are opposites of properties
we would like a system to have:

- **Spoofing** (violates authenticity)
- **Tampering** (violates integrity)
- **Repudiation** (violates ... non-repudiation)
- **Information disclosure** (violates confidentiality)
- **Denial of service** (violates availability)
- **Elevation of privilege** (violates authorization)

Spoofing

some examples

- spoofing a process on the same machine (create a file before the process creates it itself and put specific content in it, create a pipe before the process creates it, create a trojan "su" and change the PATH, name a process "explorer", ...)
- spoofing a file (create a file in the local directory, create a link and swap it between the victim checking and accessing it, ...)
- spoofing a machine on a network (ARP spoofing, IP spoofing, DNS spoofing, DNS compromise, IP redirection, ...)
- spoofing a person (create an alternate account and use someone else's display name, take over a real account, ...)
- spoofing a role (create an account with a relevant name, e.g. "IT Support Staff", ...)

Tampering

some examples

- tampering with a file (attacker modifies a file he owns and on which the victim relies, modifies a file the victim owns, possibly located on a server, . . .)
- tampering with memory (modify victim's code in memory, modify data provided to the victim's API after the security checks are performed, for instance in a pass by reference argument, . . .)
- tampering with a network (redirect the flow of traffic to the attacker's host, modify data flowing over the network, . . .)

Repudiation

some examples

- repudiating an action (claim he did not click, claim he did not receive an email, claim to be a victim of fraud, . . .)
- attacking the logs (notice there are no logs, manipulate the logs to confuse its interpretation, e.g. send "</html>" in the data if logs are displayed as a webpage)

Information disclosure

some examples

- against a process (extract secrets from error messages such as passwords typed as username, . . .)
- against a data store (inappropriate or missing file ACLs, bad database permissions, protection by obscurity, crypto keys stored unprotected on disk or in memory, revealing filenames, information in unprotected temp files, swapspace, booting from different OS to bypass ACLs, . . .)
- against a data flow (read data from the netwok, possibly using a MITM network attack, traffic analysis, analysis of DNS requests, . . .)

Denial of service

some examples

- against a process (consume all memory, all CPU, . . .)
- against a data store (fill up the filesystem, generate large amounts of requests to the data store)
- against a data flow (consume network resources)

Elevation of privilege

some examples

- against a process by corrupting the process (send inputs that the code doesn't handle properly, gain access to read/write memory)
- through missed authorization checks
- through buggy authorization checks
- through data tampering (modifies bits on disk to do things that would normally not be allowed)

STRIDE-per-element

	S	T	R	I	D	E
External Entity	x		x			
Process	x	x	x	x	x	x
Data Flow		x		x	x	
Data Store		x	?	x	x	

Figure 8: STRIDE per element

STRIDE-per-interaction

#	ELEMENT	INTERACTION	S	T	R	I	D	E
1	Process (Contoso)	Process has outbound data flow to data store.	x			x		
2		Process sends output to another process.	x		x	x	x	x
3		Process sends output to external interactor (code).	x		x	x	x	
4		Process sends output to external interactor (human).			x			
5		Process has inbound data flow from data store.	x	x			x	x
6		Process has inbound data flow from a process.	x		x		x	x
7		Process has inbound data flow from external interactor.	x				x	x
8	Data Flow (com- mands/ responses)	Crosses machine boundary		x		x	x	
9	Data Store (database)	Process has outbound data flow to data store.		x	x	x	x	
10		Process has inbound data flow from data store.			x	x	x	
11	External Interactor (browser)	External interactor passes input to process.	x		x	x		
12		External interactor gets input from process.	x					

Figure 9: STRIDE per interaction

CVE

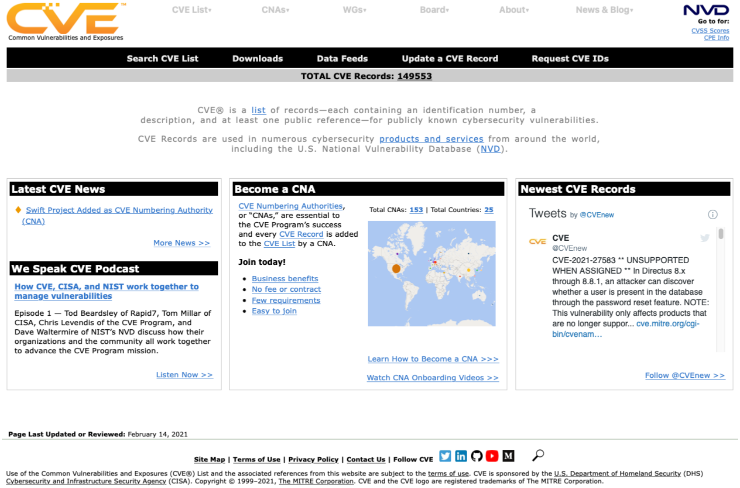
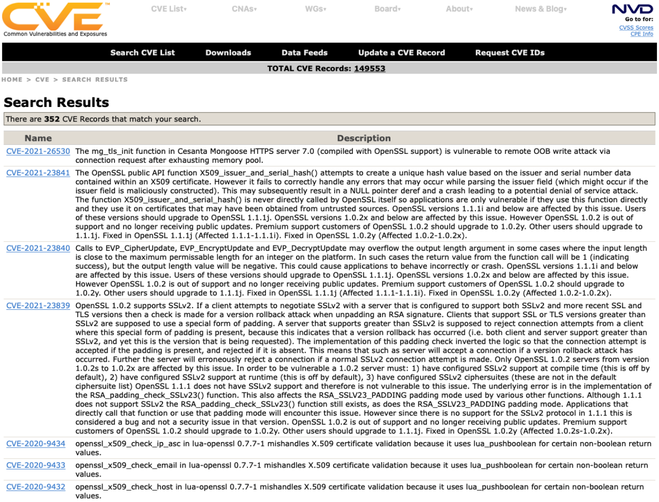


Figure 10: https://cve.mitre.org

CVE


CVE List CNAs WGs Board About News & Blog

NVD
Go to for:
CVSS Scores
CPE Info

Common Vulnerabilities and Exposures

Search CVE List | Downloads | Data Feeds | Update a CVE Record | Request CVE IDs

TOTAL CVE Records: 149553

HOME > CVE > SEARCH RESULTS

Search Results

There are **352** CVE Records that match your search.

Name	Description
CVE-2021-26530	The mg_tls_init function in Cesanta Mongoose HTTPS server 7.0 (compiled with OpenSSL support) is vulnerable to remote OOB write attack via connection request after exhausting memory pool.
CVE-2021-23841	The OpenSSL public API function X509_issuer_and_serial_hash() attempts to create a unique hash value based on the issuer and serial number data contained within an X509 certificate. However it fails to correctly handle any errors that may occur while parsing the issuer field (which might occur if the issuer field is maliciously constructed). This may subsequently result in a NULL pointer deref and a crash leading to a potential denial of service attack. The function X509_issuer_and_serial_hash() is never directly called by OpenSSL itself so applications are only vulnerable if they use this function directly and they use it on certificates that may have been obtained from untrusted sources. OpenSSL versions 1.1.1i and below are affected by this issue. Users of these versions should upgrade to OpenSSL 1.1.1j. OpenSSL versions 1.0.2x and below are affected by this issue. However OpenSSL 1.0.2 is out of support and no longer receiving public updates. Premium support customers of OpenSSL 1.0.2 should upgrade to 1.0.2y. Other users should upgrade to 1.1.1j. Fixed in OpenSSL 1.1.1j (Affected 1.1.1-1.1.1i). Fixed in OpenSSL 1.0.2y (Affected 1.0.2-1.0.2x).
CVE-2021-23840	Calls to EVP_CipherUpdate, EVP_EncryptUpdate and EVP_DecryptUpdate may overflow the output length argument in some cases where the input length is close to the maximum permissable length for an integer on the platform. In such cases the return value from the function call will be 1 (indicating success), but the output length value will be negative. This could cause applications to behave incorrectly or crash. OpenSSL versions 1.1.1i and below are affected by this issue. Users of these versions should upgrade to OpenSSL 1.1.1j. OpenSSL versions 1.0.2x and below are affected by this issue. However OpenSSL 1.0.2 is out of support and no longer receiving public updates. Premium support customers of OpenSSL 1.0.2 should upgrade to 1.0.2y. Other users should upgrade to 1.1.1j. Fixed in OpenSSL 1.1.1j (Affected 1.1.1-1.1.1i). Fixed in OpenSSL 1.0.2y (Affected 1.0.2-1.0.2x).
CVE-2021-23839	OpenSSL 1.0.2 supports SSLv2. If a client attempts to negotiate SSLv2 with a server that is configured to support both SSLv2 and more recent SSL and TLS versions then a check is made for a version rollback attack when unpadding an RSA signature. Clients that support SSL or TLS versions greater than SSLv2 are supposed to use a special form of padding. A server that supports greater than SSLv2 is supposed to reject connection attempts from a client where this special form of padding is present, because this indicates that a version rollback has occurred (i.e. both client and server support greater than SSLv2, and yet this is the version that is being requested). The implementation of this padding check inverted the logic so that the connection attempt is accepted if the padding is present, and rejected if it is absent. This means that such as server will accept a connection if a version rollback attack has occurred. Further the server will erroneously reject a connection if a normal SSLv2 connection attempt is made. Only OpenSSL 1.0.2 servers from version 1.0.2s to 1.0.2x are affected by this issue. In order to be vulnerable a 1.0.2 server must: 1) have configured SSLv2 support at compile time (this is off by default), 2) have configured SSLv2 support at runtime (this is off by default), 3) have configured SSLv2 ciphersuites (these are not in the default ciphersuite list) OpenSSL 1.1.1 does not have SSLv2 support and therefore is not vulnerable to this issue. The underlying error is in the implementation of the RSA_padding_check_SSLv23() function. This also affects the RSA_SSLV23_PADDING padding mode used by various other functions. Although 1.1.1 does not support SSLv2 the RSA_padding_check_SSLv23() function still exists, as does the RSA_SSLV23_PADDING padding mode. Applications that directly call that function or use that padding mode will encounter this issue. However since there is no support for the SSLv2 protocol in 1.1.1 this is considered a bug and not a security issue in that version. OpenSSL 1.0.2 is out of support and no longer receiving public updates. Premium support customers of OpenSSL 1.0.2 should upgrade to 1.0.2y. Other users should upgrade to 1.1.1j. Fixed in OpenSSL 1.0.2y (Affected 1.0.2s-1.0.2x).
CVE-2020-9434	openssl_x509_check_ip_asc in lua-openssl 0.7.7-1 mishandles X.509 certificate validation because it uses lua_pushboolean for certain non-boolean return values.
CVE-2020-9433	openssl_x509_check_email in lua-openssl 0.7.7-1 mishandles X.509 certificate validation because it uses lua_pushboolean for certain non-boolean return values.
CVE-2020-9432	openssl_x509_check_host in lua-openssl 0.7.7-1 mishandles X.509 certificate validation because it uses lua_pushboolean for certain non-boolean return values.

Figure 11: searching for “openssl”

OpenVAS

Date	Status	Task	Severity	Scan Results: High	Medium	Low	Log	False Pos.	Actions
Thu Jan 9 03:05:08 2020	Done	Immediate scan of IP 192.168.11.137	N/A	0	0	0	0	0	

Report: Results (312 of 734)

ID: 97cc63d0-65d7-45ee-8ca8-711df1baa7dd
Modified:
Created:
Owner: admin

1 - 100 of 312

Vulnerability		Severity	QoD	Host	Location	Actions
rexec Passwordless / Unencrypted Cleartext Login		10.0 (High)	75%	192.168.11.137	512/tcp	
Samba End Of Life Detection		10.0 (High)	75%	192.168.11.137	445/tcp	
Samba 'TALLOC_FREE()' Function Remote Code Execution Vulnerability		10.0 (High)	75%	192.168.11.137	445/tcp	
PHP Multiple Vulnerabilities - Aug08		10.0 (High)	75%	192.168.11.137	80/tcp	
PHP Version < 5.2.7 Multiple Vulnerabilities		10.0 (High)	75%	192.168.11.137	80/tcp	
PHP End Of Life Detection (Linux)		10.0 (High)	75%	192.168.11.137	80/tcp	
MySQL End Of Life Detection (Linux)		10.0 (High)	75%	192.168.11.137	3306/tcp	
PostgreSQL End Of Life Detection (Linux)		10.0 (High)	75%	192.168.11.137	5432/tcp	

Figure 12: https://www.openvas.org

cyber kill chain

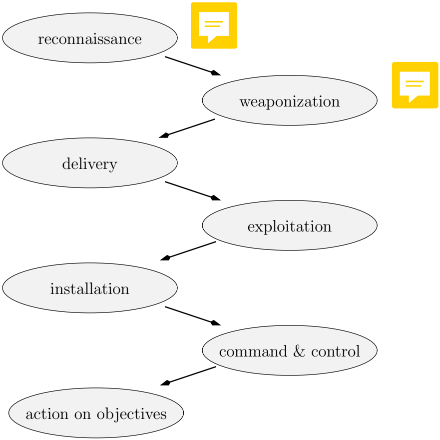
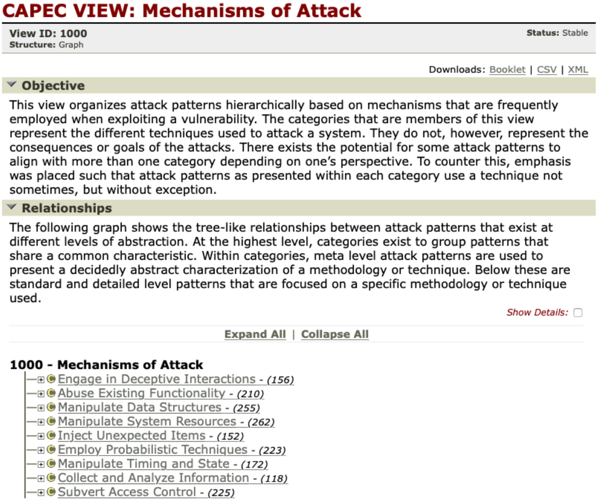


Figure 13: cyber kill chain

Common Attack Pattern Enumeration and Classification (CAPEC)

Figure 14: https://capec.mitre.org/

CAPEC



CAPEC VIEW: Mechanisms of Attack

View ID: 1000
Structure: Graph

Status: Stable

Downloads: Booklet | CSV | XML

Objective

This view organizes attack patterns hierarchically based on mechanisms that are frequently employed when exploiting a vulnerability. The categories that are members of this view represent the different techniques used to attack a system. They do not, however, represent the consequences or goals of the attacks. There exists the potential for some attack patterns to align with more than one category depending on one's perspective. To counter this, emphasis was placed such that attack patterns as presented within each category use a technique not sometimes, but without exception.

Relationships

The following graph shows the tree-like relationships between attack patterns that exist at different levels of abstraction. At the highest level, categories exist to group patterns that share a common characteristic. Within categories, meta level attack patterns are used to present a decidedly abstract characterization of a methodology or technique. Below these are standard and detailed level patterns that are focused on a specific methodology or technique used.

Show Details:

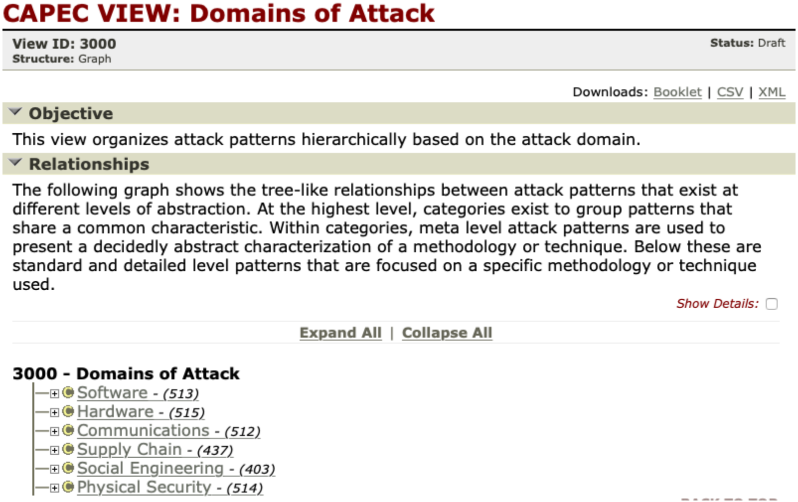
Expand All | **Collapse All**

1000 - Mechanisms of Attack

- Engage in Deceptive Interactions - *(156)*
- Abuse Existing Functionality - *(210)*
- Manipulate Data Structures - *(255)*
- Manipulate System Resources - *(262)*
- Inject Unexpected Items - *(152)*
- Employ Probabilistic Techniques - *(223)*
- Manipulate Timing and State - *(172)*
- Collect and Analyze Information - *(118)*
- Subvert Access Control - *(225)*

Figure 15: CAPEC

CAPEC



CAPEC VIEW: Domains of Attack

View ID: 3000
Structure: Graph

Status: Draft

Downloads: Booklet | CSV | XML

Objective

This view organizes attack patterns hierarchically based on the attack domain.

Relationships

The following graph shows the tree-like relationships between attack patterns that exist at different levels of abstraction. At the highest level, categories exist to group patterns that share a common characteristic. Within categories, meta level attack patterns are used to present a decidedly abstract characterization of a methodology or technique. Below these are standard and detailed level patterns that are focused on a specific methodology or technique used.

Show Details:

Expand All | **Collapse All**

3000 - Domains of Attack

- Software - *(513)*
- Hardware - *(515)*
- Communications - *(512)*
- Supply Chain - *(437)*
- Social Engineering - *(403)*
- Physical Security - *(514)*

Figure 16: CAPEC

CAPEC

Figure 17: CAPEC

CAPEC

Nature	Type	ID	Name
ChildOf	M	438	Modification During Manufacture
ParentOf	D	443	Malicious Logic Inserted Into Product Software by Authorized Developer
ParentOf	D	445	Malicious Logic Insertion into Product Software via Configuration Management Manipulation
ParentOf	D	446	Malicious Logic Insertion into Product Software via Inclusion of 3rd Party Component Dependency
ParentOf	D	511	Infiltration of Software Development Environment
ParentOf	D	516	Hardware Component Substitution During Baselining
ParentOf	D	520	Counterfeit Hardware Component Inserted During Product Assembly
ParentOf	D	532	Altered Installed BIOS
ParentOf	D	537	Infiltration of Hardware Development Environment
ParentOf	D	538	Open Source Libraries Altered
ParentOf	D	539	ASIC With Malicious Functionality

Figure 18: CAPEC

OWASP ZAP

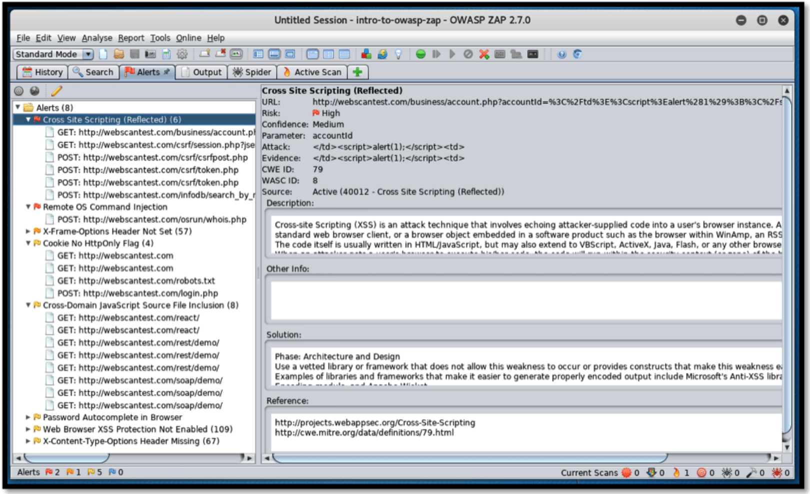


Figure 19: Open Web Application Security Project - Zed Attack Proxy

ATT&CK

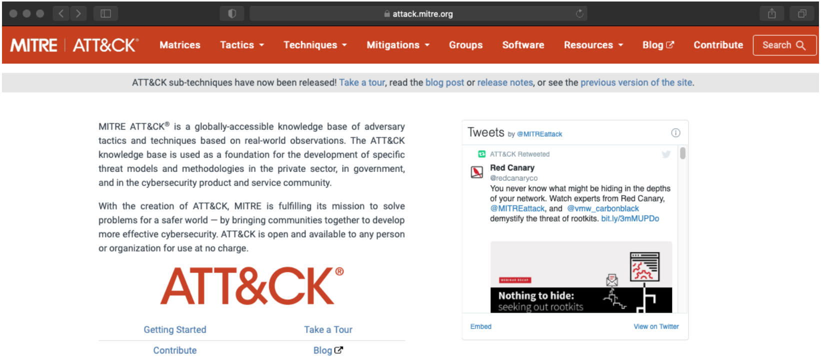
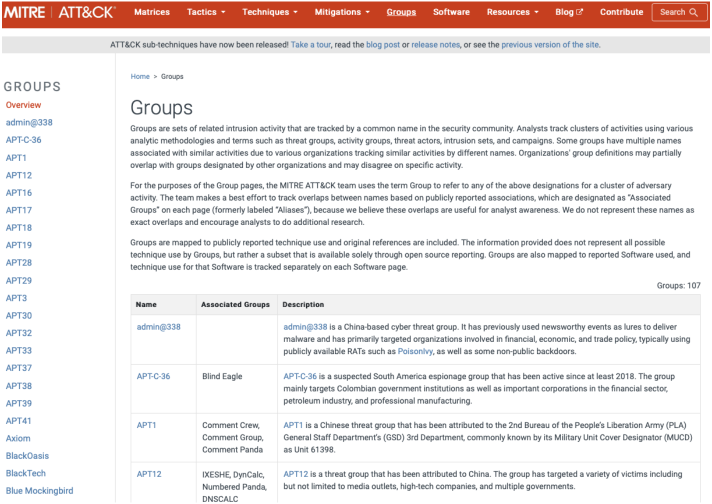


Figure 20: https://attack.mitre.org/

ATT&CK

Figure 21: ATT&CK matrix

ATT&CK



MITRE | ATT&CK® Matrices Tactics Techniques Mitigations Groups Software Resources Blog Contribute Search

ATT&CK sub-techniques have now been released! Take a tour, read the blog post or release notes, or see the previous version of the site.

GROUPS

Overview
admin@338
APT-C-36
APT1
APT12
APT16
APT17
APT18
APT19
APT28
APT29
APT3
APT30
APT32
APT33
APT37
APT38
APT39
APT41
Axiom
BlackOasis
BlackTech
Blue Mockingbird

Home > Groups

Groups

Groups are sets of related intrusion activity that are tracked by a common name in the security community. Analysts track clusters of activities using various analytic methodologies and terms such as threat groups, activity groups, threat actors, intrusion sets, and campaigns. Some groups have multiple names associated with similar activities due to various organizations tracking similar activities by different names. Organizations' group definitions may partially overlap with groups designated by other organizations and may disagree on specific activity.

For the purposes of the Group pages, the MITRE ATT&CK team uses the term Group to refer to any of the above designations for a cluster of adversary activity. The team makes a best effort to track overlaps between names based on publicly reported associations, which are designated as "Associated Groups" on each page (formerly labeled "Aliases"), because we believe these overlaps are useful for analyst awareness. We do not represent these names as exact overlaps and encourage analysts to do additional research.

Groups are mapped to publicly reported technique use and original references are included. The information provided does not represent all possible technique use by Groups, but rather a subset that is available solely through open source reporting. Groups are also mapped to reported Software used, and technique use for that Software is tracked separately on each Software page.

Groups: 107

Name	Associated Groups	Description
admin@338		admin@338 is a China-based cyber threat group. It has previously used newsworthy events as lures to deliver malware and has primarily targeted organizations involved in financial, economic, and trade policy, typically using publicly available RATs such as PoisonIvy, as well as some non-public backdoors.
APT-C-36	Blind Eagle	APT-C-36 is a suspected South America espionage group that has been active since at least 2018. The group mainly targets Colombian government institutions as well as important corporations in the financial sector, petroleum industry, and professional manufacturing.
APT1	Comment Crew, Comment Group, Comment Panda	APT1 is a Chinese threat group that has been attributed to the 2nd Bureau of the People's Liberation Army (PLA) General Staff Department's (GSD) 3rd Department, commonly known by its Military Unit Cover Designator (MUCD) as Unit 61398.
APT12	IXESHE, DynCalc, Numbered Panda, DNSCALC	APT12 is a threat group that has been attributed to China. The group has targeted a variety of victims including but not limited to media outlets, high-tech companies, and multiple governments.

Figure 22: ATT&CK hacker groups

ATT&CK

Leviathan	TEMP.Jumper, APT40, TEMP.Periscope	Leviathan is a cyber espionage group that has been active since at least 2013. The group generally targets defense and government organizations, but has also targeted a range of industries including engineering firms, shipping and transportation, manufacturing, defense, government offices, and research universities in the United States, Western Europe, and along the South China Sea.

Figure 23: ATT&CK hack group Leviathan

risk assessment

risk assessment

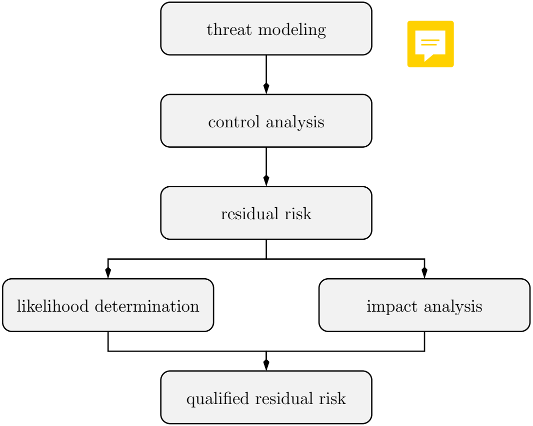


Figure 24: approach for risk assessment

qualitative risk analysis

probability \ impact	trivial	minor	moderate	major	extreme
rare	low	low	low	medium	medium
unlikely	low	low	medium	medium	medium
moderate	low	low	medium	medium	high
likely	medium	medium	medium	high	high
very likely	medium	medium	high	high	high

Figure 25: risk scoring matrix

quantitative risk analysis

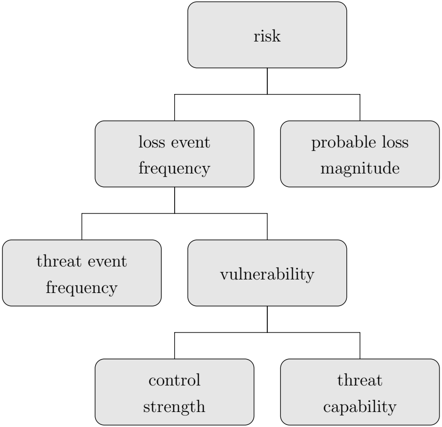


Figure 26: factor analysis for information risk (FAIR)

quantitative risk analysis

rating	description
very high (VH)	>100 times per year
high (H)	between 10 and 100 times per year
moderate (M)	between 1 and 10 times per year
low (L)	between 0.1 and 1 times per year
very low (VL)	<0.1 times per year

Figure 27: probable threat event frequency (TEF)

quantitative risk analysis

rating	description
very high (VH)	top 2% when compared against the overall threat population
high (H)	top 16% when compared against the overall threat population
moderate (M)	average skills and resources (between top 16% and bottom 16%)
low (L)	bottom 16% when compared against the overall threat population
very low (VL)	bottom 2% when compared against the overall threat population

Figure 28: threat capability (TCap)

quantitative risk analysis

rating	description
very high (VH)	protects against all but the top 2% of an average threat population
high (H)	protects against all but the top 16% of an average threat population
moderate (M)	protects against the average threat source
low (L)	only protects against the bottom 16% of an average threat population
very low (VL)	only protects against the bottom 2% of an average threat population

Figure 29: control strength (CS)

quantitative risk analysis

	control strength (CS)				
threat capacity (TCap)	VL	L	M	H	VH
VH	VH	VH	VH	H	M
H	VH	VH	H	M	L
M	VH	H	M	L	VL
L	H	M	L	VL	VL
VL	M	L	VL	VL	VL

Figure 30: vulnerability matrix

quantitative risk analysis

vulnerability (Vuln)

threat event frequency (TEF)	VL	L	M	H	VH
VH	M	H	VH	VH	VH
H	L	M	H	H	H
M	VL	L	M	M	M
L	VL	VL	L	L	L
VL	VL	VL	VL	VL	VL

Figure 31: loss event frequency matrix

quantitative risk analysis

magnitude	description
severe (SV)	10.000.000\$ <= loss
high (H)	1.000.000\$ <= loss < 10.000.000\$
significant (Sg)	100.000\$ <= loss < 1.000.000\$
moderate (M)	10.000\$ <= loss < 100.000\$
low (L)	1.000\$ <= loss < 10.000\$
very low (VL)	loss < 1.000\$

Figure 32: probable loss magnitude

quantitative risk analysis

loss can be due to...

- *"replacement":* intrinsic value of asset itself
- *"response":* cost associated with managing the incident (man-hours, logistics, ...)
- *"productivity":* organization looses (part of) its capacity to produce value
- *"fines and judgments":* legal or regulatory actions against the organization as a result of the cyber incident
- *"competitive advantage":* losses due to for instance trade secrets, or merger and acquisition plans getting released
- *"reputation":* external perception that the organization is unethical, staff or leadership is incompetent, ...

quantitative risk analysis

loss event frequency (LEF)

probable loss magnitude (PLM)	VL	L	M	H	VH
SV	H	H	C	C	C
H	M	H	H	C	C
Sg	M	M	H	H	C
M	L	M	M	H	H
L	L	L	M	M	M
VL	L	L	L	M	M

Figure 33: risk magnitude matrix

result: "low" (L), "moderate" (M), "high" (H), or "critical" (C)

risk management

risk management

possible options for reducing a risk

- reduce the impact
  e.g. reduce the information disclosure impact when an attacker breaks into a network by moving the most sensitive data to a separate network
- reduce the probability
  e.g. reduce the probability of spoofing an authorized user by implementing two-factor authentication (for remote access)

impact

probability

	trivial	minor	moderate	major	extreme
rare	low	low	low	medium	medium
unlikely	low	low	medium	medium	medium
moderate	low	low	medium	medium	high
likely	medium	medium	medium	high	high
very likely	medium	medium	high	high	high

Figure 34: risk scoring matrix

risk management

alternative options for managing a risk

- *risk reduction:* systems or activities are modified, or security controls added, such that risk is reduced to acceptable level
- *risk avoidance:* systems or activities are (temporarily) stopped or modified in such a way that risk is removed
- *risk transfer:* risk liability or responsibility is shifted to another organization by purchasing an insurance, outsourcing an information service, ...
- *risk acceptance:* when risk is sufficiently low, or when taking risk is necessary from business point of view, we may simply accept the risk

output: risk treatment plan

conclusions

conclusions

Figure 35: questions or comments ?