Chapter 7

Thursday, August 17, 2023 10:53 AM

Understanding Attack Frameworks

attack frameworks to identify tactics, techniques, and procedures

TTP

Cyber Kill Chain

Kill chain

- identify target
- dispach resources to target
- decide to attack and give order
- destroy target

Cyber kill chain

1. **Reconnaissance**
   research, identify, select targets
2. **weaponization**
   Malware delivered in payload
3. **Delivery**
   payload transmit to target

4. Exploitation
   weapon delivered and activated

5. Installation
   install RAT or backdoor
   maintain persistance

6. Command and Control
   send beacon out through internet
   give full access to attacker

7. Actions on Objectives
   begin taking action (like install ransomware, data exfiltration...

Diamond Model of Intrusion Analysis

Adversary
   email @, usernames or forums,
   APT membership...

Capabilities
   malware, exploits, and other tools

Infrastructure

domain names, email @, IP addrs used

Victim

name, email, network identifiers

every event has an adversary that uses capability across an infrastructure against a Victim

MITRE ATT&CK

MITRE ATT&CK - adversarial tactics, techniques and Common Knowledge

Knowledge base of tactics and techniques used in real-world attacks

attacks

tactics = why they are doing what they are doing

techniques = how objective complete

tactics
- init access
- execution
- persistence
- privilege escalation
- defense evasion
- credential access
- discovery
- lateral movement
- collection and exfiltration
- command and control

tactics columns, items techniques

Identifying Network Attacks

DoS versos DDoS

DoS = 1 attacker on 1 target

DDoS = 2+ pc against 1 target

DDoS - 2nd pt 3

Resource exhaustion - overload sys resources and prevent legit users from accessing resources

Operational technology OT - methods used to monitor and manage industrial control systems

SYN Flood Attacks

disrupts TCP handshake

send barrage of SYN packets but never complete

systems often have limits on half open connections, will close off new connections denying legit users

Spoofing

spoofing - one person or entity impersonates as someone

as someone

- email: change sender/reply-to addr
- IP
- MAC: associate different MAC to NIC

On-Path Attacks

man-in-the middle

active interception/eavesdropping

separate pc to forward, manipulate, interrupt, or listen to traffic between 2 pcs

Sophisticated on-path

create multiple secure connections

1 w/ p1

1 w/ p2

receive data, decrypt & store, encrypt and send

delay can be strong indicator of attack

- of attack

Certificates will not be from legit CA, so users will get warnings

SSH man in the middle attacker try to change key fingerprints

SSH will send alert if changed

Man-in-the-browser

proxy trojan that infects vulu browsers

capture browser session data

keystrokes, data, form grabbing

Secure sockets layer stripping

SSL stripping - changes HTTPS connection to HTTP

Layer 2 attacks

data link layer transfers frames between systems

switches

ARP Poisoning Attacks

ARP poisoning - misleads pcs of switches about actual MAC address of system

ARP resolves IP of system to their MAC

ARP Messages

ARP request - broadcasts IP address to see who it is

ARP reply - PC w/ IP address responds w/ its MAC

pc that sends request caches MAC addr

in many OS all systems that hear reply also cache MAC addr

ARP will believe any reply

can easily create ARP reply

can easily create AR... packets w/ spoofed MAC

ARP on-path Attacks

eavesdrop, redirect network traffic, insert malicious code

traffic usually goes

user → switch → router

but after ARP poisoning traffic is redirected to attacker

user → switch → attacker → router

IP forwarding to send to router

ARP DoS Attacks

attacker can send ARP reply w/ MAC address for default gateway which is IP for router that provides path out of net

can do this for all pcs on net so none of them can

net so none of the
connect to internet

MAC Flooding

attack against a switch that
tries to overload with MAC
for each port

typically 1:1 device:port

send large # of fake MAC
to same port

switch will run out of mem
to store all MAC and reverts
to fail-open state - becomes
a hub

attacker can then use packet
sniffer on any hub to
see all traffic

flood guard - limits amount of mem
used to store MAC addresses for

Flood guard

used to store MAC addresses for each port

ex: only 132 entries per port

typically sends SNMP trap or error message

disable or refuse updates to port meaning it only uses current entries

MAC Cloning

changing system's MAC addr to another

ex: fool ISP into thinking new router is the old one

DNS Attacks

can sometimes use reverse lookup to detect sus activity

use IP to get real name instead of forged one

DNS Poisoning Attacks

DNS Poisoning Attacks

modify or corrupt DNS data

replace stored IP for sites w/ malicious ones

many DNS servers have DNSSEC to prevent

Pharming Attack

tries to corrupt DNS server or client

also tries to redirect to different sites

modify hosts file on Windows

URL Redirection

send traffic to another place within site or another site

Domain Hijacking

change Domain name registration without permission from owner

often use social engineering to escape

often use social engineering
w/ stolen email

Domain Reputation

determines likelihood that email is sent by legit org

DNS Sinkhole

DNS server that gives incorrect results for one or more domain names

DNS Log Files

record DNS queries

Replay Attacks and Session Replays

replay attack - replay data that was already part of comm session

capture data, modify, and impersonate one of the parties in session

replays

timestamps and sequence #s thwart

Summarizing Secure Coding Concepts

software

OWASP

nonprofit for improving software security

Code Reuse and Dead Code

Code reuse is safe and recommended
- properly tested
- less dev time

dead code - code never executed or used

Third-party Libraries and SDKs

libraries and SDKs help w/ reuse

Input Validation

Checking data validity before using it

if
- buffer overflow
- SQL injection
- DLL injection
- XSS

common checks

- proper chars
- block HTML

- block HTML
- prevent certain characters
- boundary or range checking

Client-side and Server-side Input Validation

Client = quick but vulu to attack

server = long but more secure

Can bypass client by

- disable JS
- web proxy to capture clients data and modify HTTP POST request

Other Input Validation Techniques

escape or encode HTML

ex: > ⇒ &gt

OWASP Enterprise Security API ESAPI

security tool library

Avoiding Race Conditions

race condition - two or more modules try to access same resource

race co

try to access same

internal concurrency control - process to prevent editing same resource

time of check to time of use TOCTOU

state attack

try to race OS to do something after it passes time of check but before time of use

Proper Error Handling

if errors are not handled, it can often give attacker debug info

errors should be general

detailed info should be logged

Code Obfuscation and Camouflage

rename vars, replace #, remove comments...

not recommended - security through obscurity

Software Diversity

automated software diversity use compiler to mimic compilers of mult languages

creates binary file that includes all functions as if it was written in mult lang

also adds randomness so each system acts slightly different

Outsourced code development

outsource devs to code

- make sure it works
- check for vuln code
- malicious code like backdoors/logic bomb
- lack of updates

Data Exposure

protect data at rest, transit, use

encryption

HTTP Headers

HTTP Headers

Client = HTTP requests

Server = HTTP responses

Header can have diff groups

general header group - entire message

request header group - info about browser, lang, encoding

entity header group - body of message

good OWASP headers for responses

- HTTP strict-Transport-security
  only display page if HTTPS
- content-security-policy
  defines mult sources of acceptable content - JS, css, images...
- X-Frame-Options
  if x-frames allowed

Secure Cookie

.. site. Site creates

Secure Cookie

When user visit site, site creates cookie and saves to user system

small file that can have any content

secure cookie - has secure attr set only transmit over HTTPS

NW if secure is set, many browsers dont send over HTTP

Code Signing

certificates can be used to auth and validate code

identifies author

hash verifies code has not been modified

Analyzing and Reviewing Code

common QA methods

- static code analysis

common

- static code analysis
  examines code without running
- manual code review
- dynamic code analysis
  check code as it is running
  fuzzing - use random data
- sandboxing

Software version control

tracks versions of software as its updated

Secure Development Environment

Different stages in development

- development
  isolated env
- test
- staging
- production
- QA

Database concepts

typically use SQL

Normalization

organize tables and cols to reduce redundant data and improve performance

First Normal Form

1NF

- each row is unique and identified w/ primary key

  primary key = tutorID

  each row has unique value

  can combine tables w/ composite primary key (two primary keys, 1 from each)

- related data is contained is a separate table

  of the cols include repeating

none of the cols include repeating groups

Second Normal Form

2NF

only applies to tables w/ composite primary key

must be in 1NF

non-primary key cols are completely dependent on composite

if only dependent on one of composite, not 2NF

Ex: Book + Author

publisher col is dependent on book, can only have 1 Authors can work w/ multiple publishers so not dependent

Third Normal Form

dependencies

Third Normal Form

3NF helps reduce redundancies

must be in 2NF

all cols that aren't primary keys are only dependent on primary key

SQL Queries

orgs use SQL to query db to render data requested by user or site

SQL Injection Attacks

attacker enters add data into webpage form to generate SQL statements

; = line end

-- = comment

can enter into name field

Ethan'; SELECT * FROM customers;--

Ethan ; --

this may change the expected statement to

SELECT * FROM customers WHERE AUTHOR = 'Ethan';)

so the injected statement could be ran

the -- will comment out the original ;

' or '1' = '1'; --

SELECT * FROM customers WHERE name = '' or '1' = '1'; --'

this may do

SELECT ... WHERE NAME = ''

SELECT ... WHERE '1' = '1'

attacks typically start by generating db

attacks typically start by ...
errors to learn about db

Protecting Against SQL Injection Attacks

input validation

Stored procedures - group of SQL statements that execute as a whole

parameterized stored procedure - accepts statements as input rather than taking input directly from UI

Provisioning and Deprovisioning

providing and removing access to users

deprovisioning can be delete/disable account

Also applies to apps

refers to what permissions the ...

refers to what permissions the app has on OS

camera, GPS, ...

deprovision can be removing data when app deleted

Integrity Measurement

quality of the code

how well it was tested

Web server logs

log activity on server

normal & abnormal activity

Using Scripting For Automation

SIEM uses scripts or alerts

ex: send email to admin

SOAR tools as well

Devops

- Automated courses of action
  ... changes don't break something

- Automated courses
  verify changes don't break something
- Continuous monitoring
  monitors for compliance
- Continuous validation
  revalidates code after changes
- Continuous integration
  after validation
  merging code
- Continuous delivery
  code is auto released to staging
- Continuous deployment
  changes auto deployed

Identifying Malicious code and Scripts

primary indicator for attacks using scripts is logs

Powershell

task-based cli that uses cmdlets

- .bat files for command prompt
- .ps1 files for powershell

Model com

.ps—

Microsoft component object Model COM
Windows Management Instrumentation
WMI

also allow admin to query/manage
Linux & Mac systems

can run powershell scripts in mem
w/o ever saving file to disk

cmdlets use verb-noun

InvokeCommand

verb	noun
. Get	. Command
. Add	. service
. Test	. Location
. Remove	. Process

Bash

run script w/

bash mytest.sh

sh mytest.sh

./mytest.sh

/bin/bash mytest.sh

/bin/sh mytest.sh

Python

mostly interpreted and can be run using python shell

can also compile into .pyc

sometimes .pyo .pyw

Macros

typically using VBA to manipulate Windows macros

Visual Basic for Applications VBA

internal prog lang for MS apps

event-driven

ex: key press

Macros typically disabled by default

OpenSSL

software to implement SSL and TLS

Linux terminal

software to

accessible in Linux terminal

SSH

technically a shell env

can run with SSH

OpenSSH - suite of tools to simplify ssh

Identifying Application Attacks

Zero-Day Attacks

unknown to trusted sources

erratic, unexpected, behavior

Memory Vulnerabilities

need to use secure mem management in code

Memory Leak

a bug in a pc app that causes app to use more and more mem the longer it runs

could crash OS

typically caused when app reserves mem but never releases

indicator = system running slower until rebooted

can view leaks w/ tools like task manager

Buffer overflows and Buffer overflow Attacks

when app receives more input or diff input than it expects

error exposes system mem

normally an app has access to specific mem, called buffer, getting access to more mem allows attacker to insert malicious code

ex: app expects 15 char name

ex: app expects 15 char [illegible]

using more would cause buffer overflow

NO-operation NOP

used to guess where the new overflow memory locations are

NOP slide or NOP sled

intel processors use hex 90 or x90 as NOP command

attacker writes long string of x90 instructions followed by mal code

When PC is exec code from mem and gets to NOP, it goes to next mem location

with multiple NOPs it slides through all of them and executes code at the end of them

code at the end of mem

if attacker can get pc to exec code in mem anywhere in NOP slide, malcode gets exec

indicater of buffer overflow is list of NOP instructions

Integer Overflow

when app recieves # too big for it to handle

reserves specific # of bits to store variable,

Pointer/Object Dereference

pointer is mem address of variable or object

Java dereference = set to null + garbage collection

c++ & c# dereference allow read/write to pointer

location

to pointer
can modify original mem location

C++ or C# trying to use a null object can cause memory leak

need null checks

Other Injection Attacks

more than SQL

Dynamic Link Library Injection

apps use DLLs - compiled set of code that app can use

DLL injection - injectr DLL into system's mem and causes it to run

attack DLL to running process, allocate mem within running process, connects DLL in mem,

process, connects DLL in mem,
execs fone

Lightweight Directory Access Protocol injection

LDAP - specifies formats to query db of objects like users, pcs, ...

injection possible if web app used to query LDAP-based db

similar to modifying SQL statements

Extensible Markup Language Injection

supports user defined tags

XML commonly used to transfer data

ex: insert xml into email accept field

& executed

tiela

indicator = creation of unwanted accounts

Directory Traversal

specific type of injection attack that attempts to access file by including full directory path

ex: ../../etc/passwd

can insert rm -rf

Cross-site scripting

XSS

allow attackers to inject scripts into web pages

reflected XSS or non-persistant

starts w/ malicious email

click link and sends HTTP

Click link and sends request to server

servers sends it back to user in response

Stored XSS or persistant

Stored in db or trusted location by web app

web app retrieves malicious code later

Cross-site Request Forgery

XSRF or CSRF - attacker tricks user into performing action on site

craft URL

google.com/search?q=Success

CAPTCHA is good defense

XSRF token - required in any page form

XSRF token - requires ... that includes form

large random # generated on each form

when submit, includes token along w/ other data

web app verifies that token in request is same as web form

Server-Side Request Forgeries

exploit how server processes external info

ex: site reads data from external URL

Attacker modifies external site

api data, db, files

Client-Side Request Forgeries

attacker inject code into client-side webpage after server crafted it and sent to user

cookies

inject code into existing cookies web app expects to read on client

Driver Manipulation

drivers interact w/ hardware/software

shimming - make it appear that older drivers are compatible

driver shim - add code that can be run instead of driver

attackers can create shims and trick OS into using it

OS redirects call to driver to

OS redirects can

scum

AI and ML

AI is intelligence in systems

ML is concept of learning off data/rules

AI and ML in Cybersecurity

- Spam
- IDS/IPS SOAR
- Risk prediction

Adversarial AI

fool AI models by supplying it deceptive input

Tainted Data for ML

try to cause inconsistent results

Security of ML Algorithms

proprietary

Security of MD5

algos used to be proprietary

Assessment Notes

1. Spike in net traffic from mult sources

   DDoS

2. CPU usage was fine but then averaging 98% for last two min

   Resource exhaustion

3. SSH remote host ID has changed

   on-path attack

4. Malware infected but not email browsing internet all day

browsing internet

DNS servers

web servers only show activity on itself

5. App crashes several times reporting mem error + malcode buffer overflow

6. Display generic error, log detailed info

7. At minimum (can only choose 1) should have server-side val

8. Code signing = validate app has not been modified

9. Dynamic code analysis for ...ability of app

4. Dynamic testing reliability of app

input val is done in the app

not a testing method

10. version control = document changes

11. Normalization = reduce query times

12. Store procedures = no SQL injections

13. ' or '1' = '1'; -- = SQL injection

14. CSRF or XSRF = testing URL with params ?

15. Powershell script uses verb noun

invoke-command