KENNESAW STATE
UNIVERSITY

Module 6:

Security issues in Physical Systems (Sensor, Actuators, Hardware) - Session Hijacking, Social Engineering

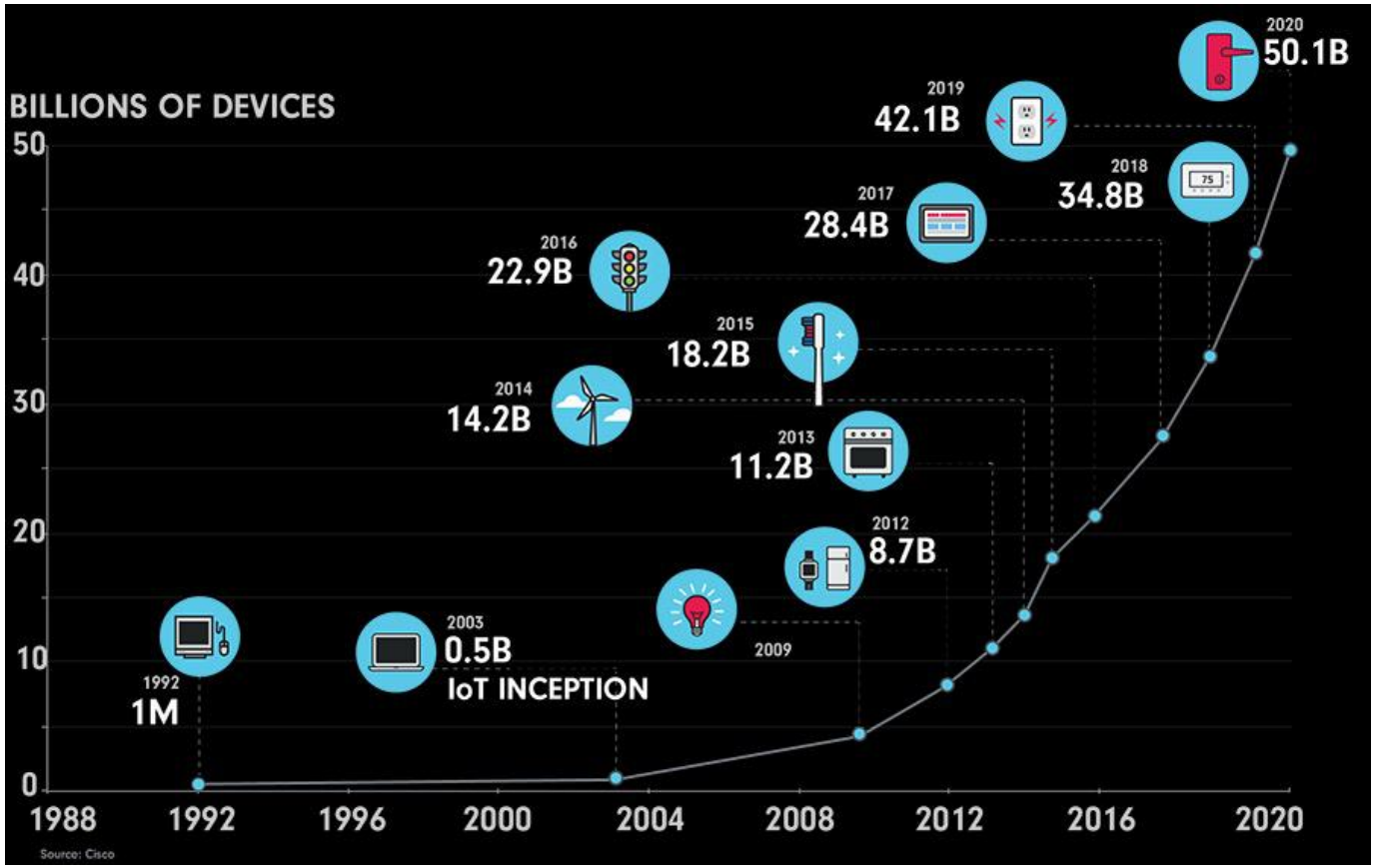
Dr. Maria Valero

Agenda

- Trend in IoT Devices
- IoT Communication
- IoT Attack Surfaces
- Defensive Security Measures
- Real-life Case
- Session Hijacking
- Social Engineering

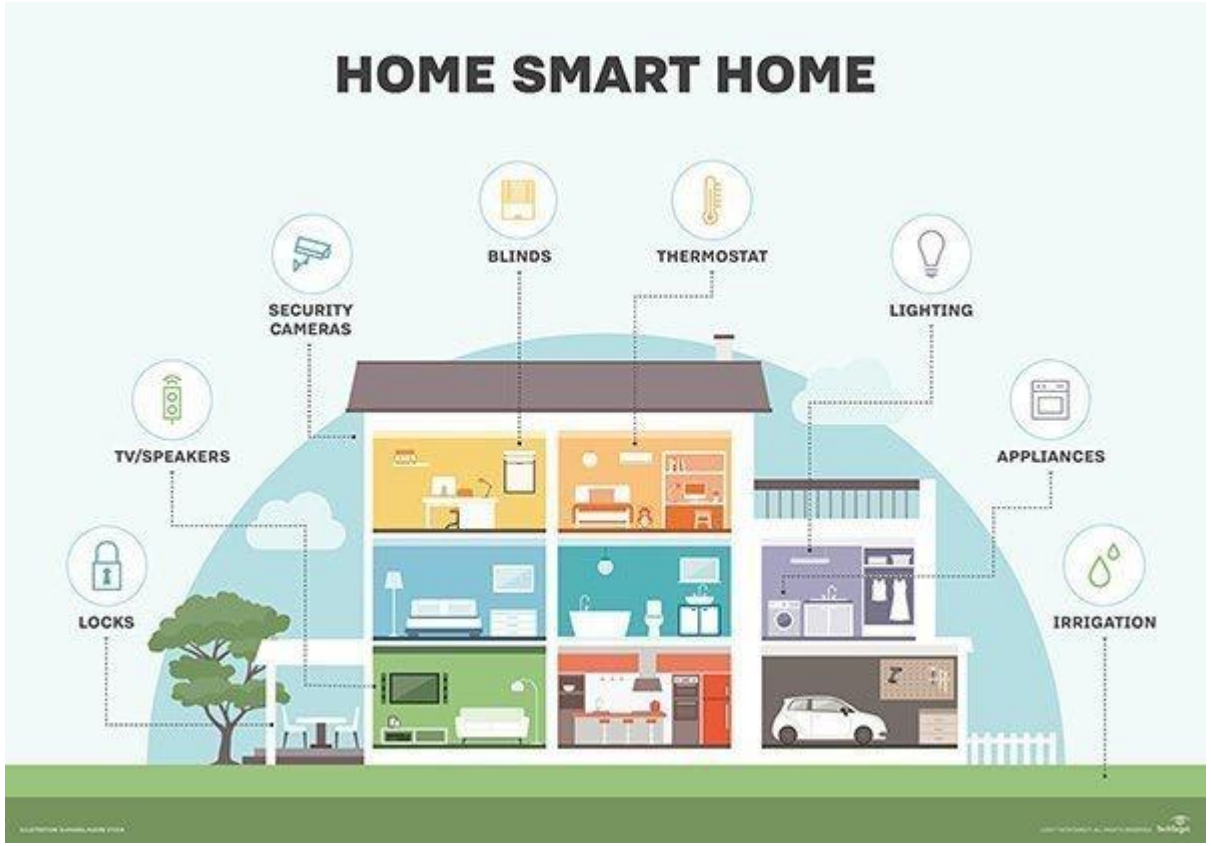
Trend in IoT Devices

- Number of IoT Devices has surpassed the number of humans on the planet
- Industries:
  - Personal/Consumer
  - Healthcare
  - Automotive
  - Manufacturing
  - Etc.



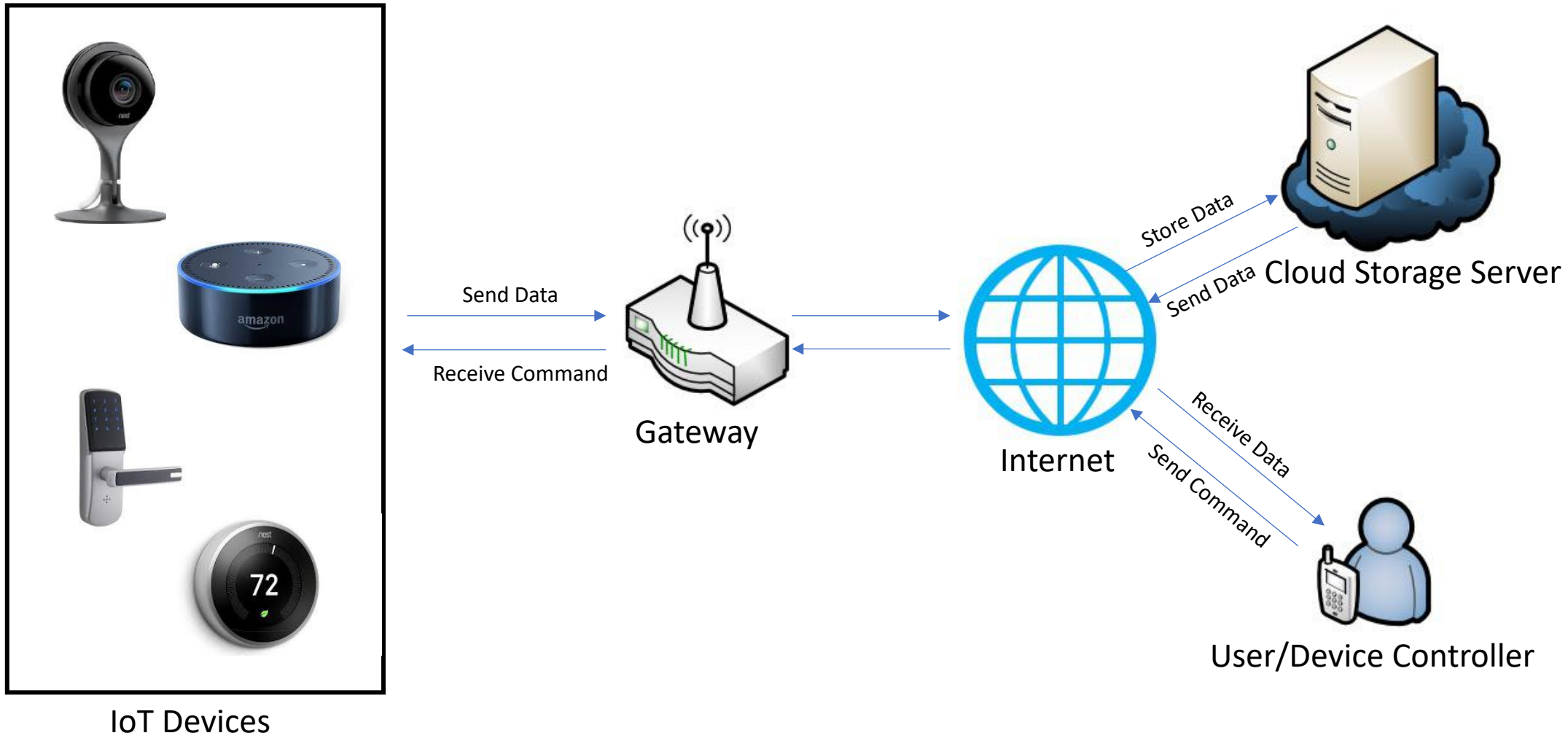
Home IoT Devices

- Average number of devices per person:
  - **8** devices per person (Cisco VNI 2018)



https://internetofthingsagenda.techtarget.com/definition/smart-home-or-building

IoT Communication



IoT Attack Surfaces (1)

Attack Surface	Vulnerability
Ecosystem Access Control	• Implicit trust between components • Enrollment security • Lost access procedures
Device Memory	• Cleartext usernames • Cleartext passwords • Third-party credentials
Device Physical Interfaces	• User CLI • Admin CLI • Privilege escalation
Device Web Interface	• SQL Injection • XSS • Weak Passwords
Device Firmware	• Hardcoded credentials • Sensitive information disclosure • Encryption keys

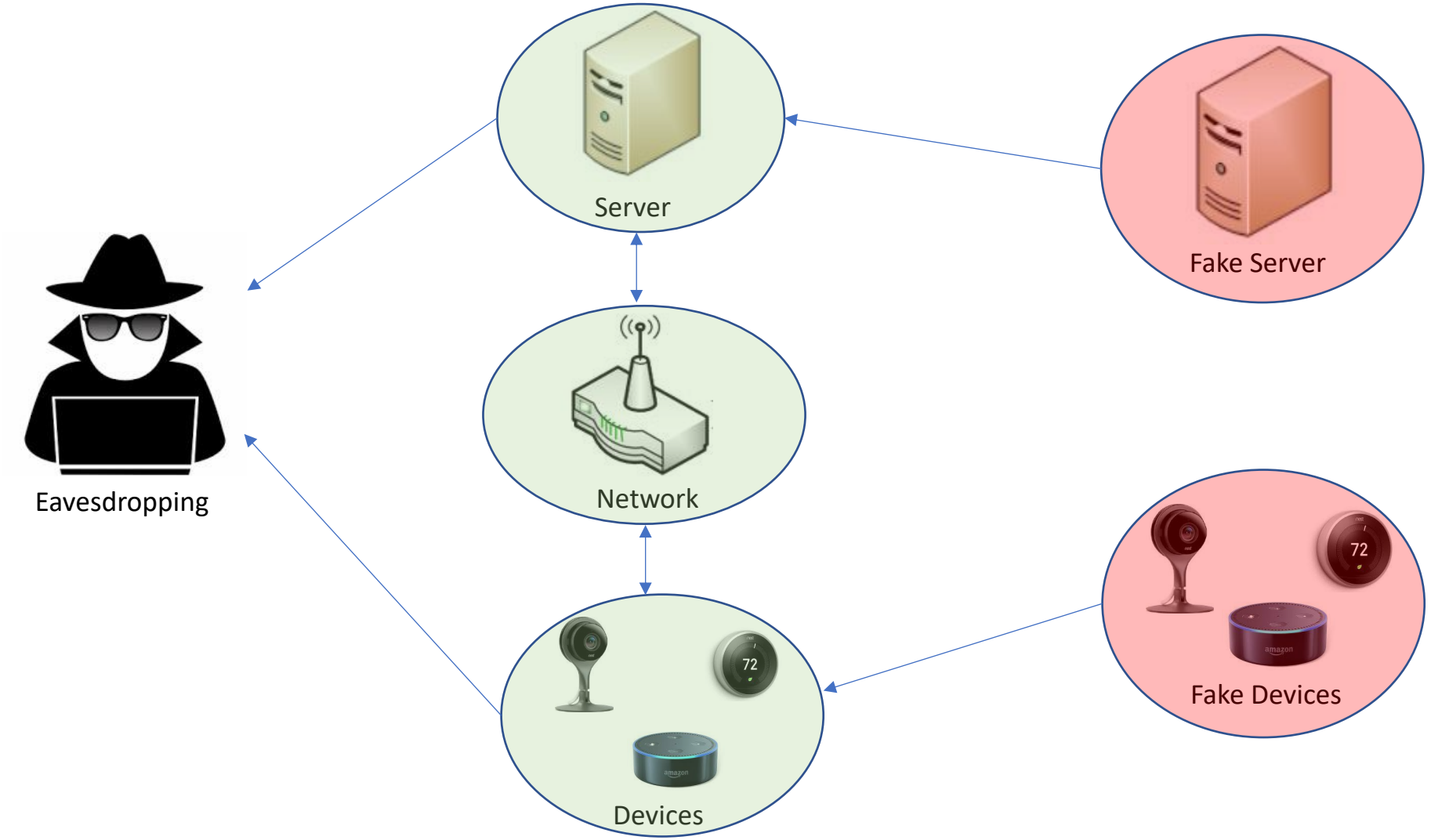
IoT Attack Surfaces (2)

Attack Surface	Vulnerability
Device Network Services	• Denial of Service • Buffer Overflow • Poorly implemented encryption
Administrative Interface	• SQL Injection • Account lockout • Two-factor authentication
Local Data Storage	• Unencrypted data • Data encrypted with discovered keys • Lack of data integrity checks
Cloud Web Interface	• SQL Injection • Weak passwords • Username enumeration
Third-party Backend APIs	• Unencrypted PII sent • Device information leaked • Location leaked

IoT Attack Surfaces (3)

Attack Surface	Vulnerability
Update Mechanism	• Update sent without encryption • Updates not signed • Missing update mechanism
Mobile Application	• Implicitly trusted by device or cloud • Insecure data storage • Insecure password recovery mechanism
Vendor Backend APIs	• Inherent trust of cloud or mobile application • Weak access controls • Weak authentication
Ecosystem Communicaiton	• Health checks • Ecosystem Commands • Pushing updates
Network Traffic	• LAN • LAN to Internet • Short range

Defensive Security Measures (1)



Defensive Security Measures (2)

Category	IoT Security Consideration
Insecure Web Interface	• Disallow weak user passwords • Provide an account lockout mechanism • Test interface for SQL injection, XSS, CSRF vulns
Insufficient Authentication/Authorization	• Require strong passwords for authentication • Implement two-factor authentication • Force password expiration after a certain date
Insecure Network Services	• Ensure all devices operate with minimal ports active • Ensure devices do not make network ports or services available to internet via UPnP • Review required network services for vulnerabilities
Lack of Transport Encryption	• Ensure traffic is encrypted between system components • Ensure SSL/TLS implementations are updated and configured properly
Privacy Concerns	• Ensure only minimal amount of PII is collected from consumers • Ensure only non-sensitive data is analyzed • Ensure data retention policy is in place

Defensive Security Measures (3)

Category	IoT Security Consideration
Insecure Cloud Interface	• Ensure all cloud interfaces are reviewed for vulnerabilities • Ensure any cloud-based web interface disallows weak passwords • Ensure all cloud interfaces use transport encryption
Insecure Mobile Interface	• Ensure that any mobile application disallows weak passwords • Ensure that any mobile application has an account lockout mechanism • Implement two-factor authentication for mobile applications
Insufficient Security Configurability	• Ensure password security options are made available (e.g. Enabling 20 character passwords or enabling two-factor authentication) • Ensure encryption options are made available (e.g. Enabling AES-256) • Ensure secure logging is available for security events
Insecure Software/Firmware	• Ensure all system devices have update capability and can be updated quickly when vulnerabilities are discovered • Ensure update files are encrypted and that the files are also transmitted using encryption
Poor Physical Security	• Ensure the device is produced with a minimal number of physical external ports (e.g. USB ports) • Ensure the firmware of Operating System can not be accessed via unintended methods such as through an unnecessary USB port • Ensure the product is tamper resistant

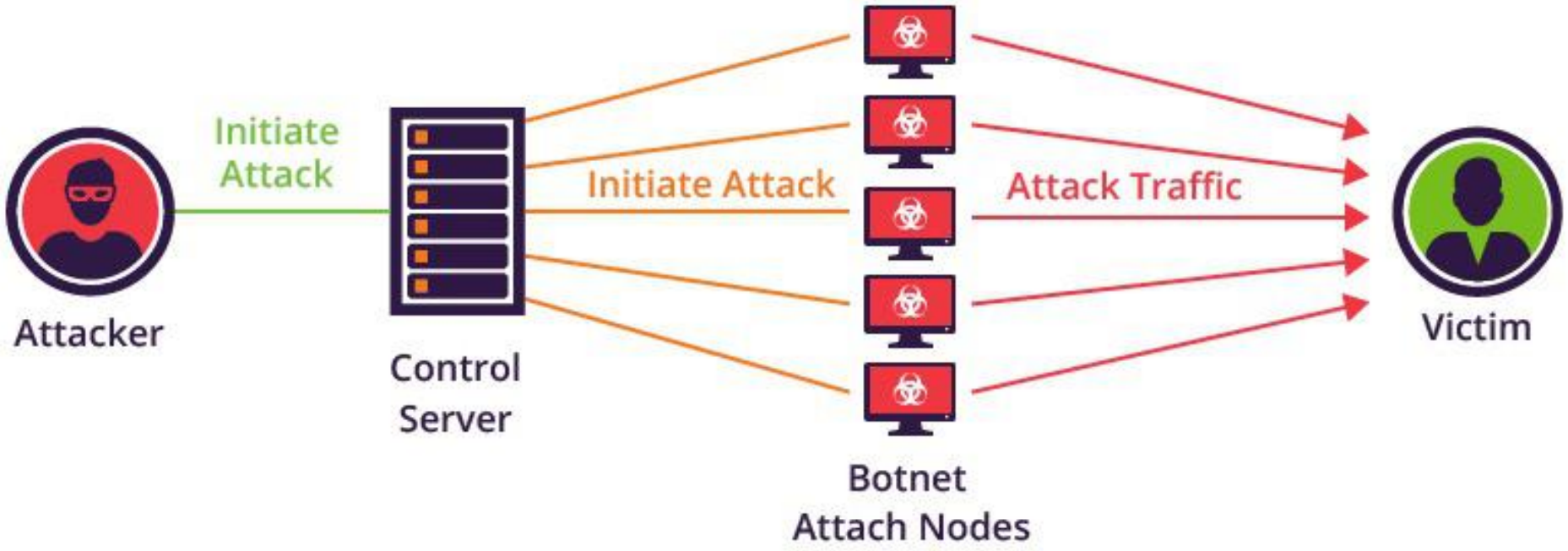
Case: Dyn Botnet DDoS Attack (1)

- DDoS Attack in October, 2016 → Target: DNS provider **Dyn**
  - DDoS attack was staged and launched from IoT devices using the Mirai malware
- **Mirai was designed for two main purposes:**
  - Find and infect IoT devices to grow the botnet
  - Participate in DDoS attacks based on commands received by remote Command and Control (C&C) infrastructure
- **Mirai operates in three stages:**
  1. Infect the device
  2. Protect itself
  3. Launch attack

Case: Dyn Botnet DDoS Attack (2)

- **Stage 1:** Scan for IoT devices that are accessible over the Internet
  - Primarily scans for ports **22, 23, 5747,** etc. that are open
  - Can be configured to scan for others
- Once connected → brute-forces usernames and passwords to login to the device
- Use the device to scan networks looking for more IoT devices

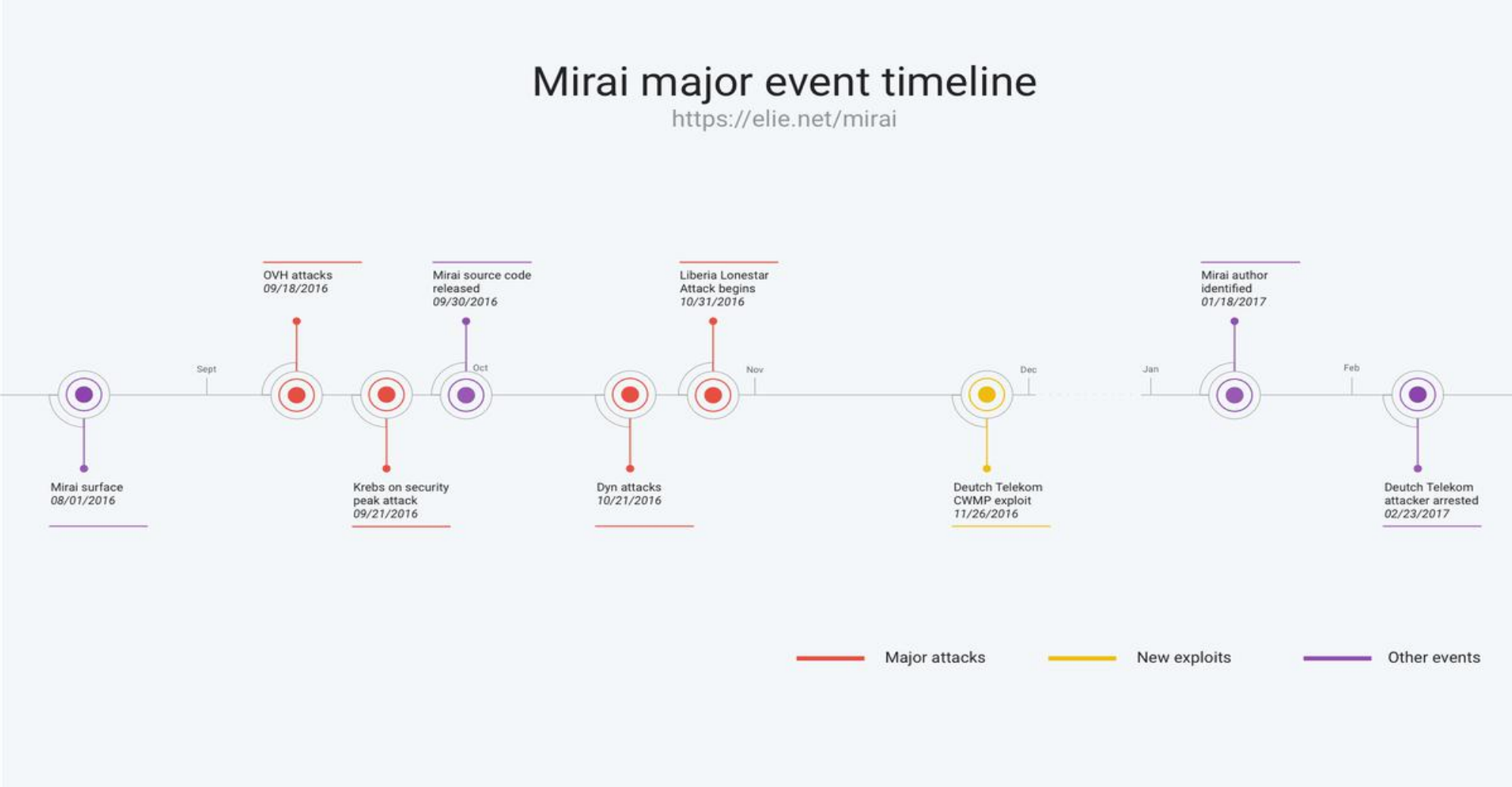
Case: Dyn Botnet DDoS Attack (3)



Case: Dyn Botnet DDoS Attack (4)

- **Stage 2:** Protect itself
  - Kill other process running on infected device (SSH, Telnet, HTTP) to prevent owner from gaining remote access to device while infected
  - **Note:** Rebooting the device can remove the malware, but it can become infected again
- **Stage 3:** Launch attack
  - Infected device launches different types of attacks
  - HTTP floods, SYN floods, etc. → DDoS-based attacks
- **Note: Mirai contained a list of known networks in the U.S. to avoid attacking → U.S. Postal Service, Department of Defense

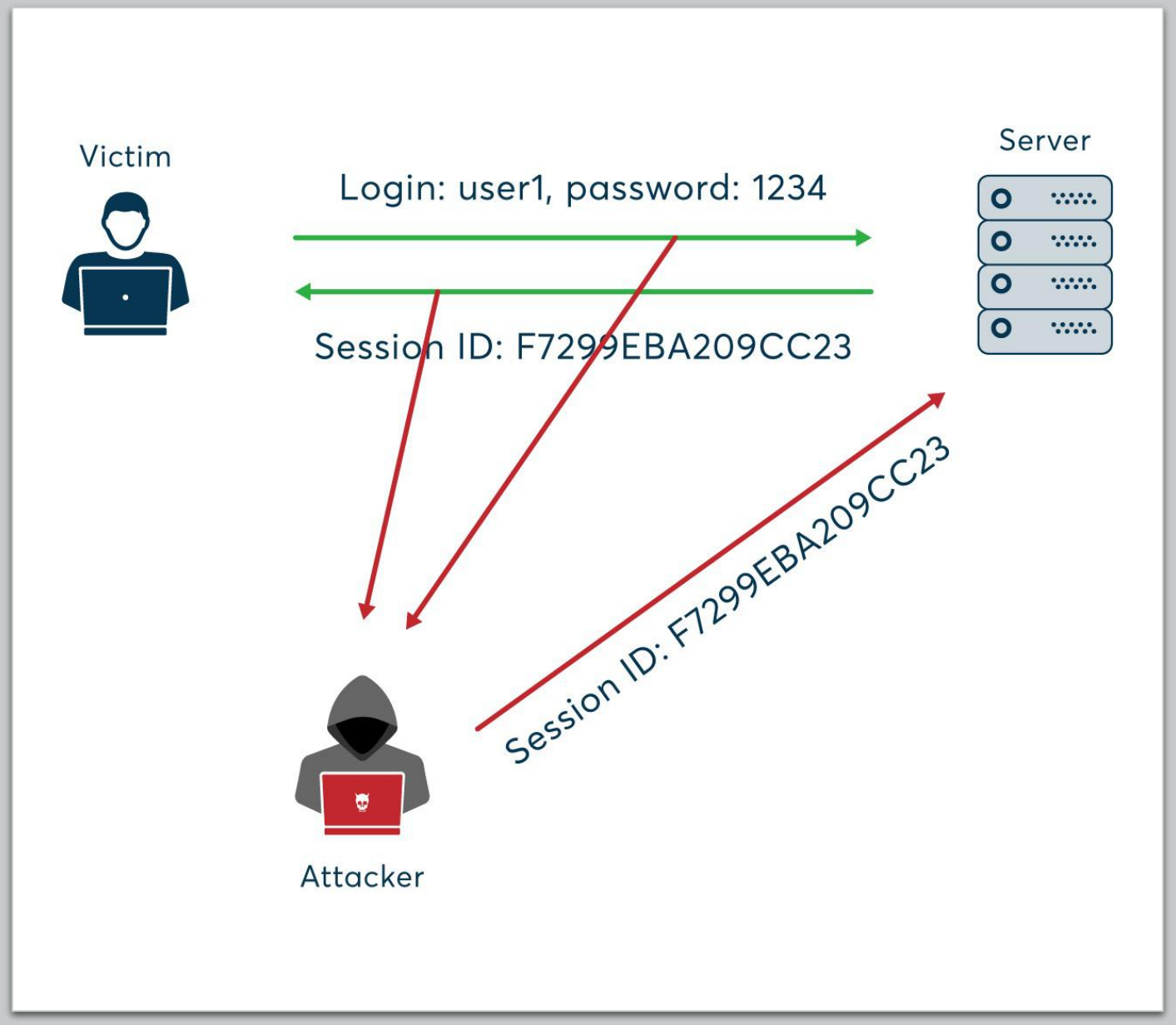
Case: Dyn Botnet DDoS Attack (5)




Session Hijacking

Session Hijacking (1)

- Session hijacking, also known as TCP session hijacking, is a method of taking over a web user session by surreptitiously obtaining the session ID and masquerading as the authorized user.

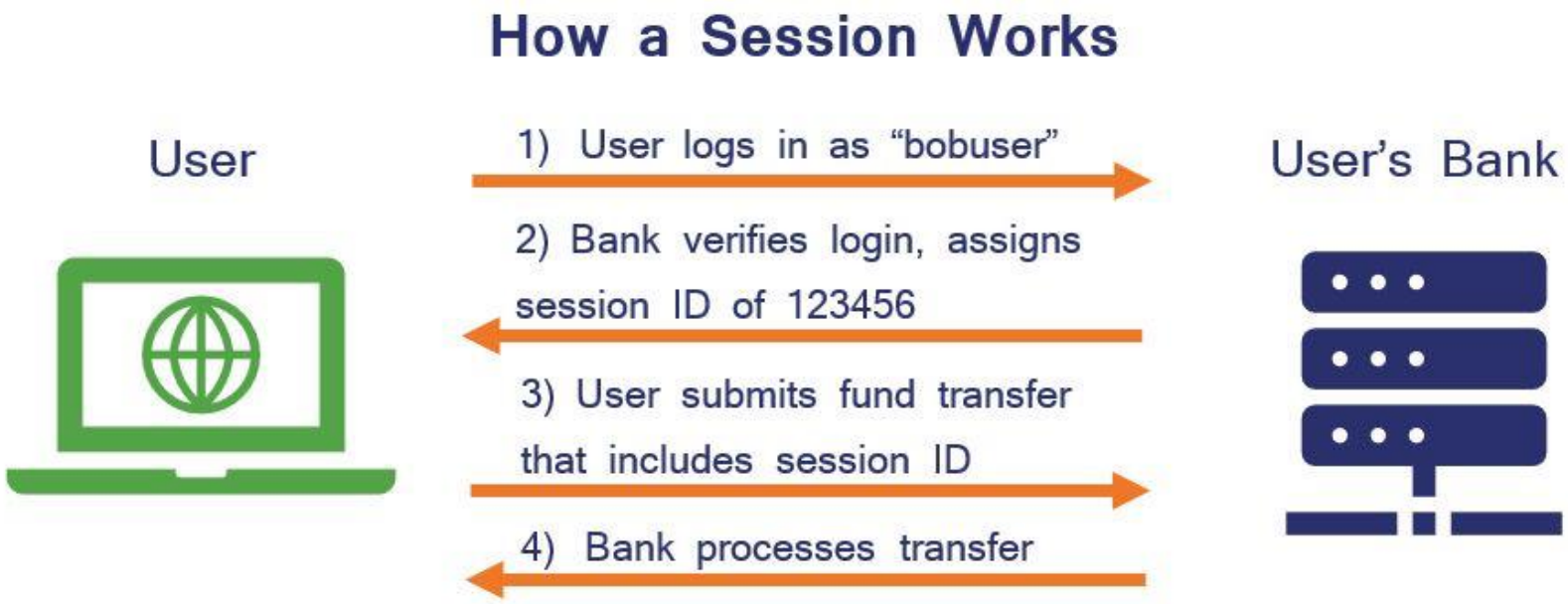


Session Hijacking (2)

- Once the user's session ID has been accessed, the attacker can masquerade as that user and do anything the user is authorized to do on the network.
- One of the most valuable byproducts of this type of attack is the ability to gain access to a server **or IoT device** without having to authenticate to it. Once the attacker hijacks a session, they no longer have to worry about authenticating to as long as the communication session remains active. The attacker enjoys the same server access as the compromised user because the user has already authenticated to the server prior to the attack.

What is Session?

- A session is a series of interactions between two communication end points that occurs during the span of a single connection



How does session hijacking works?

- Session Sniffing
  - The attacker uses a sniffer, such as Wireshark, or a proxy, such as OWASP Zed, to capture network traffic containing the session ID between a website and a client. Once the attacker captures this value, he can use this valid token to gain unauthorized access.
- Predictable sessions token ID
  - Many web servers use a custom algorithm or predefined pattern to generate session IDs. The greater the predictability of a session token, the weaker it is and the easier it is to predict. If the attacker can capture several IDs and analyze the pattern, he may be able to predict a valid session ID.

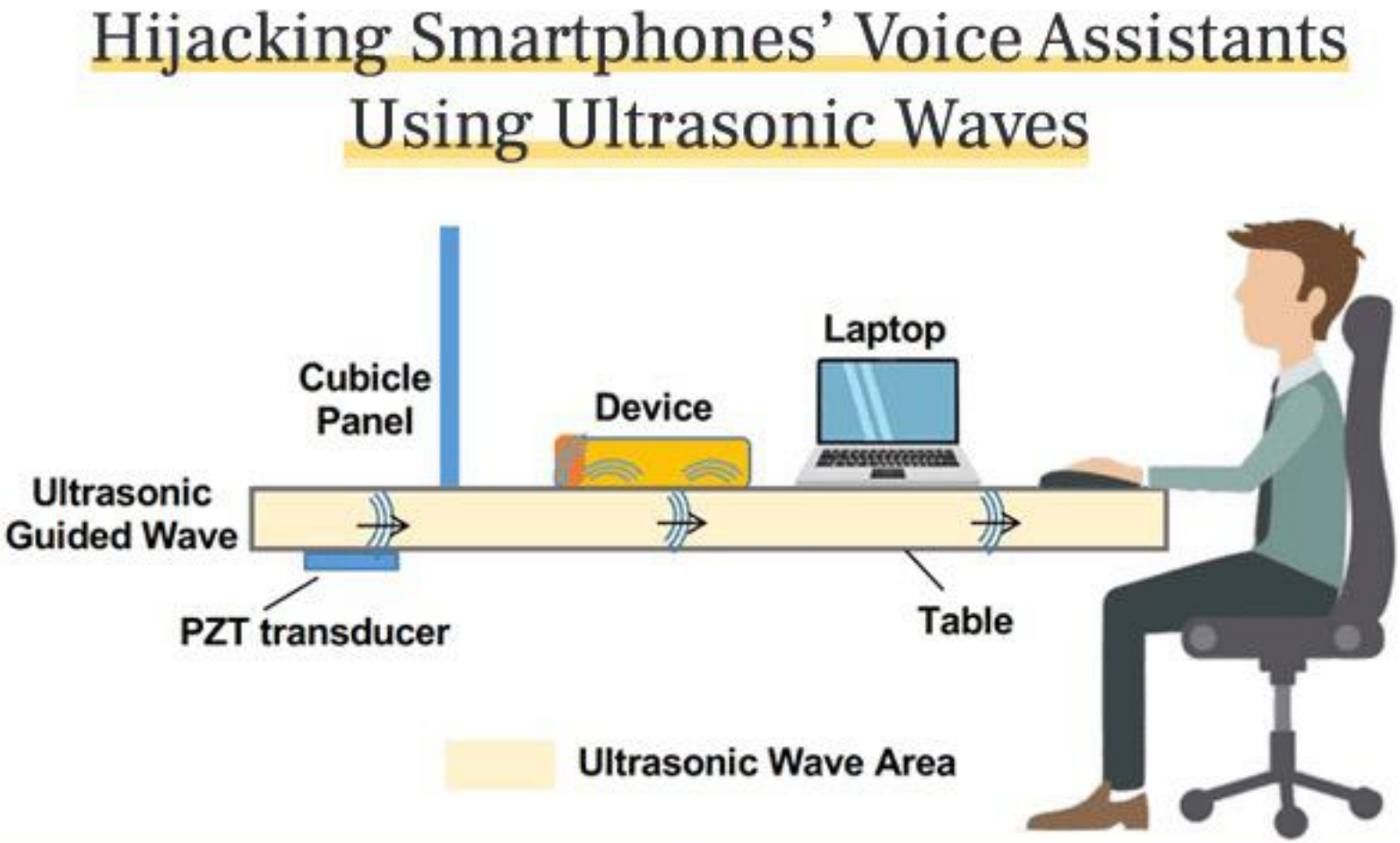
How does session hijacking works? (2)

- **Cross-site Scripting**
  - Cybercriminals exploit server or application vulnerabilities to inject client-side scripts into web pages. This causes the browser to execute arbitrary code when it loads a compromised page. If HttpOnly isn't set in session cookies, cybercriminals can gain access to the session key through injected scripts, giving them the information, they need for session hijacking.
- **Session fixation attacks**
  - This technique steals a valid session ID that has yet to be authenticated. Then, the attacker tries to trick the user into authenticating with this ID.

Can session hijacking compromise our voice assistants?

Yes... It is possible

The novel attack exploits the nonlinear nature of MEMS microphone circuits to transmit malicious ultrasonic signals — high-frequency sound waves that are inaudible to the human ear — using a $5 piezoelectric transducer that's attached to a table surface. What's more, the attacks can be executed from as far as 30 feet.



https://surfingattack.github.io/

Social Engineering Attacks in IoT

Social Engineering Attacks (1)

- In the context of information security, social engineering is the psychological manipulation of people into performing actions or divulging confidential information.

Social Engineering Attacks (2)

1. **Phishing**: tactics include deceptive emails, websites, and text messages to steal information.
2. **Spear Phishing**: email is used to carry out targeted attacks against individuals or businesses.
3. **Baiting**: an online and physical social engineering attack that promises the victim a reward.
4. **Malware**: victims are tricked into believing that malware is installed on their computer and that if they pay, the malware will be removed.
5. **Pretexting**: uses false identity to trick victims into giving up information.
6. **Quid Pro Quo**: relies on an exchange of information or service to convince the victim to act.
7. **Tailgating**: relies on human trust to give the criminal physical access to a secure building or area.
8. **Vishing**: urgent voice mails convince victims they need to act quickly to protect themselves from arrest or other risk.
9. **Water-Holing**: an advanced social engineering attack that infects both a website and its visitors with malware.

Social Engineering Attacks on IoTs (1)

- A security risk associated with IoT which is often overlooked is the increased vulnerability to ***social engineering*** attacks .
- IoT devices often hold the trust of users as they belong to a family of devices which they have been able to safely use for years, such as cars, phones, and television sets. The trust relationship between users and IoT devices makes them an effective avenue for social engineering attacks because users are more likely to accept information received from them without question.

Social Engineering Attacks on IoTs (2)

- Examples -> The threat agents triggers a message to all the smart TVs, which is displayed the next time they start:

Your TV requires a software upgrade.

For your security, it will stop working in 60 minutes, until upgraded.

Please go to www.example.com/smartTVupgrade and download the patching software, and run it from any Windows computer on the same network as this TV

Social Engineering Attacks on IoTs (3)

- Examples -> Once compromised the espresso Thing displays the message that says:

50% off coffee pods! 1 Day Sale!

Please enter your espresso account PIN to make your purchase

- Enter PIN into the machine like you normally do to order, which opens the trusted local storage and releases the account data. Unfortunately, at this point the account data is captured through a man-in-the-middle