BITS Pilani presentation

BITS Pilani
Pilani Campus

Paramananda Barik
CS&IS Department

BITS Pilani
Pilani Campus

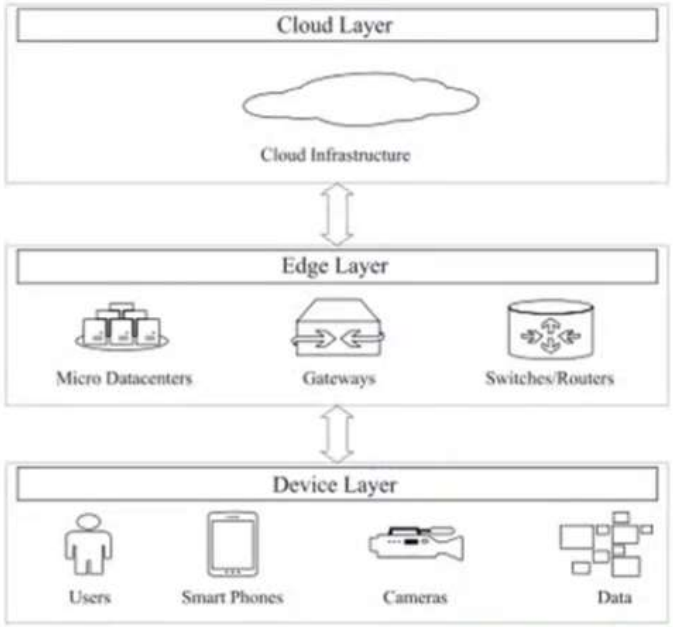
SEZG586/SSZG586,
Security and Privacy in Edge Computing

Lecture No.13

A Conceptual Framework for Security and Privacy in Edge Computing

Edge computing devices - specific attributes such as:

- heterogeneity
- low latency
- location awareness
- spatial distribution
- mobility



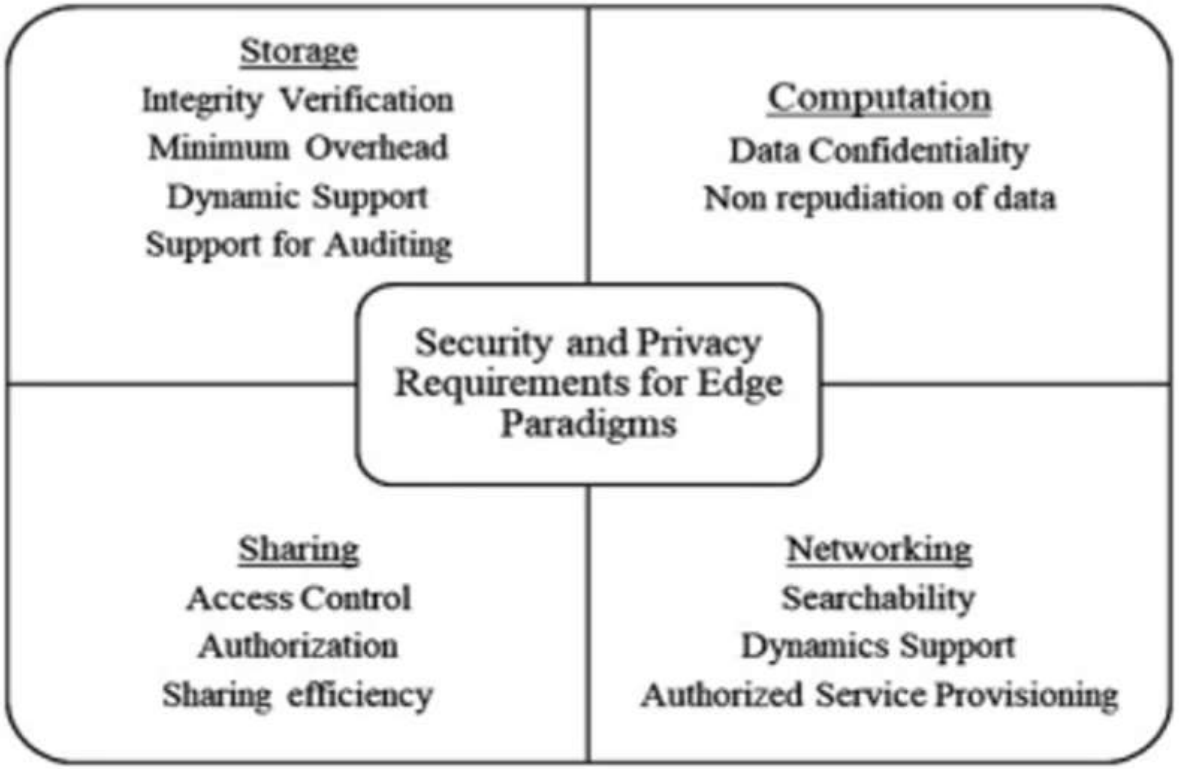
Difference between edge computing and cloud computing

Features	Cloud computing	Edge computing
Computational capacity	High	Medium to low
Size and operating mode	Centralized large servers	Distributed smaller servers
Applications	Delay tolerant, computationally intensive	Low latency, real-time operation, high QoS
Fronthaul/backhaul communication overhead	High	Low
Mobility support	Low	High
Management	Service provider	Local business
Deployment	Requires complicated deployment planning	Ad hoc deployment/minimal or no planning
User device	Computers, limited mobile devices	Mobile smart wearable devices
Network access type	WAN	LAN/WLAN

Overview of Security, Privacy, and Threat in Edge Computing

The principal services offered by edge computing

- storage
- computation
- data sharing
- networking



Network Security

Edge devices

- distributed across the network
- cost of maintenance is high

therefore

Configurations by the network administrator

Network management information

- must be secured
- isolated from the normal data flow

Data Storage Security

If the data storage in edge platform is outsourced

- tough to ensure data integrity
- possibility for data loss and data modifications

Data abuse by the attackers is easy

Hence,

- data storage auditing is needed

Encryption techniques

- ensure integrity, verifiability, and confidentiality of the data

Data Computation Security

Data computation performed by the edge servers and devices should be

- secured
- verifiable

Computations can be made secure by using data encryption techniques

- Prevents data visibility to any hackers/attackers

Microdata centers have the provisions to off-load

- mechanism to verify the computation results
- establish trust between the two entities

End Devices Security

End devices can be easily tampered

- less powerful
- limited access

Potential harm that can be done by an attacker:

- create rogue node to retrieve essential network management data
- change normal behavior by corrupting the device data
- increase the frequency of data access by sending fake information
- propagate false data across the network

Access Control

Edge platform is distributed and decentralized

- good access control policy acts as a defensive shield

Access control helps in the following:

- realizing the interoperability
- collaboration among microdata centers that are provided by different service providers and are separated across different geo-locations

Intrusion Detection

Intrusion detection techniques help in the following:

- identifying malicious data entries
- detect device anomalies

Used to investigate and analyze the behavior of devices in the network

Provide methods to perform packet inspection:

- early detection of denial-of-service attacks, integrity attacks, and data flooding

Privacy in Edge Computing

Privacy in edge computing:

- user privacy
- data privacy
- location privacy

User privacy deals with protecting the privacy of user's credentials and their frequency of accessing the data.

Data privacy helps in avoiding unauthorized access of the network by hackers and preserve the integrity of the network.

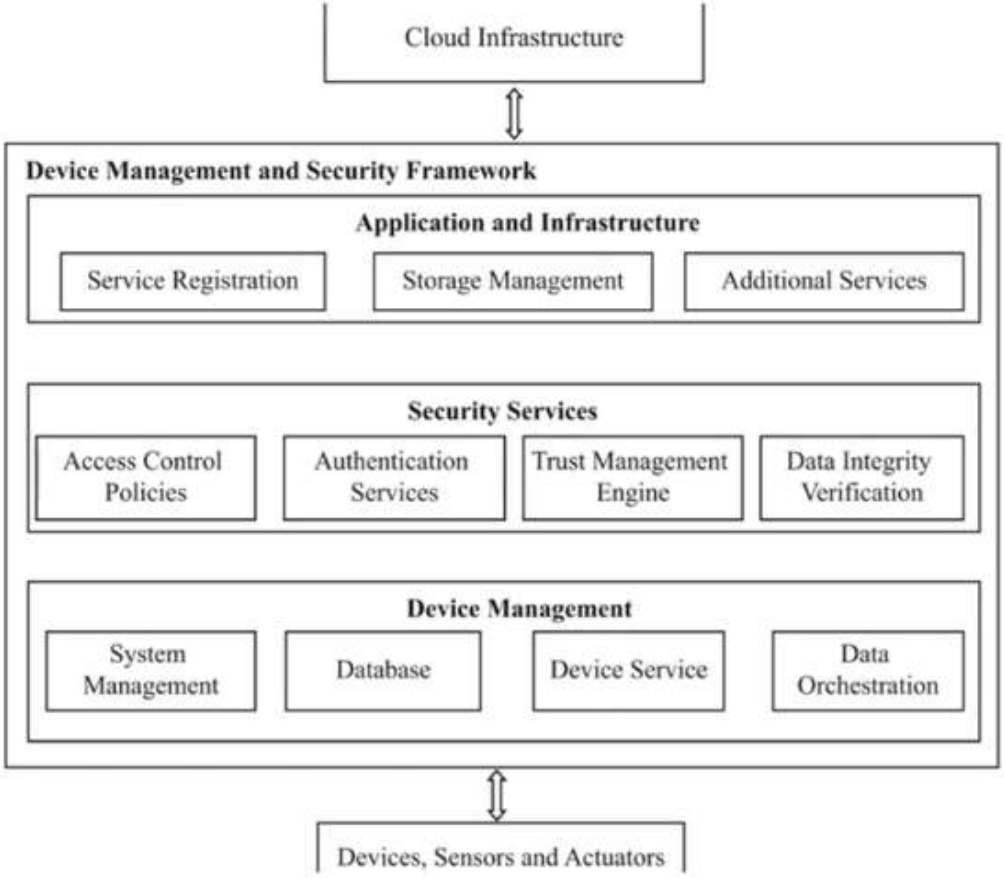
Location privacy is required since the edge data centers may be spatially distributed

Ensuring location privacy without affecting the computational overhead is of primary importance

Nature of Threats in Edge Computing

Components	Possible threats
Data	Data replication
	Data manipulation
	Data deletion
	Illegal data access
	Eavesdropping
Network	Denial of service
	Man-in-the-middle
	Physical damage
	Service manipulation
	Rogue server
	Privacy leakage
	Jamming
	Congestion
	Black hole
Communication	Data loss
	Data breach
	Sniffing
	Message replay
	Impersonation
Services	Service manipulation
	Rogue infrastructure
	Privacy leakage
	Insecure APIs
Devices	Physical damage
	Misuse of resources
	Flooding
	Power failure

Framework for Security and Privacy in Edge Computing



Device Management

Primary functionalities of the device management layer:

- manage the end devices
- create database
- manage the services provided to the end device
- perform data orchestration operations

Device management layer takes care of:

- status of the end devices
- access control
- maintaining integrity of the data stored
- provide the necessary data from service providers

Security Services

Security services layer performs:

- creating, updating, and maintaining the access control policies.
- ensure authentication of devices, service providers, and the data that is communicated.
- makes use of a trust management engine
  - governs the privacy of the overall system
  - generates rules to implement intrusion detection mechanism

Application Infrastructure

Responsible for

- keeping track of the network-wide application interactions
- governs the authentication policies for accessing the network devices
- service provisioning details of neighboring data centers
- decision of whether cloud support is required or not

Security and Privacy Challenges in Edge Computing

- The added security mechanism should not increase the computation overhead and hamper the storage space required for other system operations.
- Edge computing mechanisms use cache management techniques which are prone to side channel attacks; hence, care should be taken to prevent leakage of private and sensitive data.

Security and Privacy Challenges in Edge Computing

- For applications that stream data continuously for monitoring and management purposes, due to increased number of devices, the volume data to be processed is typically large. To detect the anomaly, packet filtering mechanisms are needed which might require additional memory and processing power.

- Due to the increased number of devices, implementing a common access control policy becomes a tedious task. Accounting for mobility of devices is also important.

Security Threats

1) Weak Computation Power
2) Attack Unawareness
3) OS and Protocol Heterogeneities
4) Coarse-Grained Access Control

Mirai virus - 65 000 IoT devices - first 20 h - August 2016 - few days later - turned into botnets - launch distributed denial of service (DDoS) - shutting down over 178 000 domains

IoTReaper and Hajime – infect more than 378 million IoT devices in 2017

Weak Computation Power

Computation power of an edge server is relatively weaker.

Similarly, compared to general-purpose computers, edge devices have more fragile defense systems;

Attack Unawareness

General purpose computers have UI

IoT devices do not have user interfaces (UIs)

Therefore, a user may have limited knowledge about the running status of a device

OS and Protocol Heterogeneities

General purpose computers use:

- Standard Os
- Communication protocols such as POSIX

Edge devices have different OSes and protocols without a standardized regulation

Coarse-Grained Access Control

Access control model in cloud:

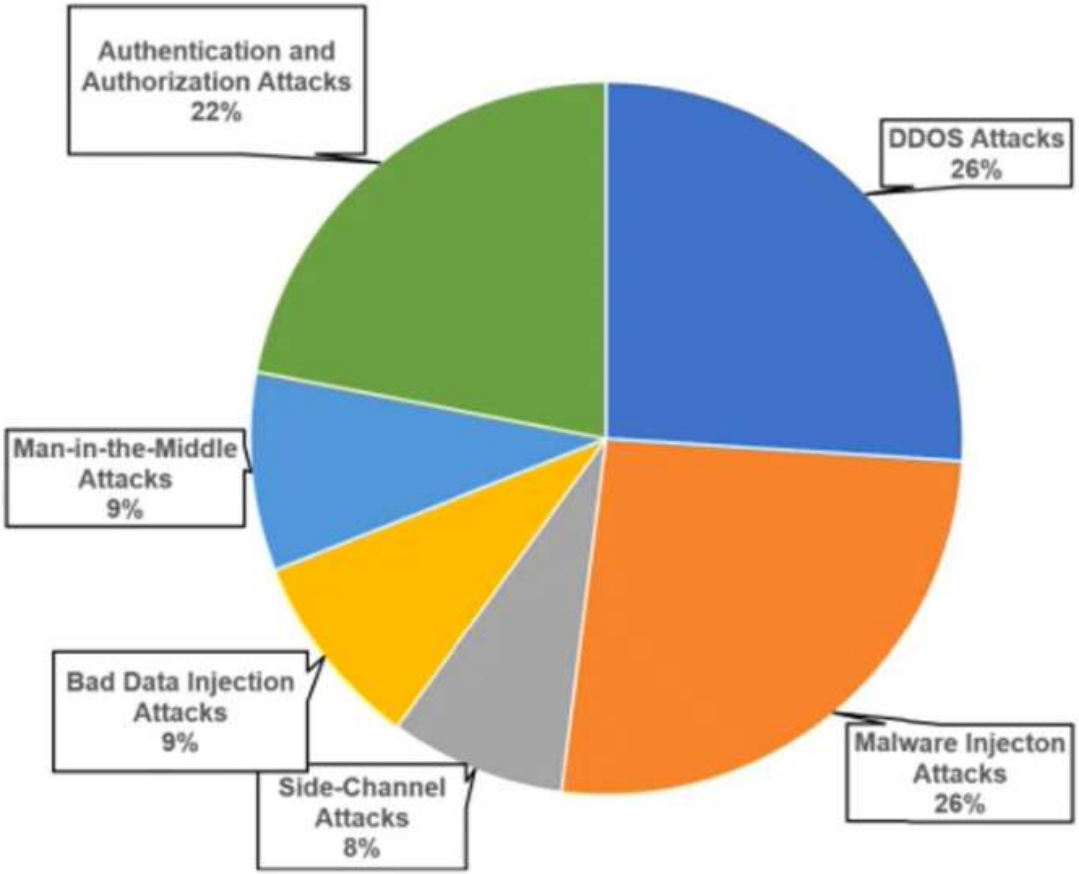
four types of permissions:

- No Read & Write
- Read Only
- Write Only
- Read & Write

Access control in edge should be able to answer:

such as “who can access which sensors by doing what at when and how”

Types of attacks targeting edge computing infrastructure



Security threats and attacks faced by edge computing

Threats mainly due to

- the design flaws
- misconfigurations
- implementation bugs

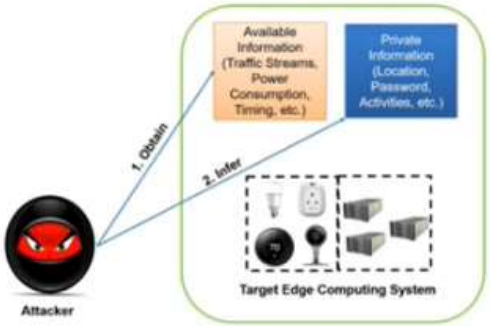
Defense mechanisms

- Detection based
- Prevention based

Types of attacks

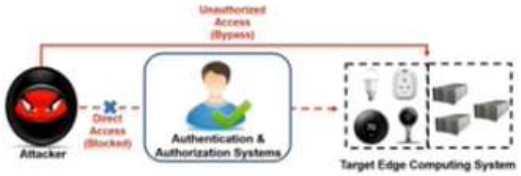
DDoS Attacks

Side-Channel Attacks



Malware Injection Attacks

Authentication and Authorization Attacks



DDoS Attacks and Defense Mechanism

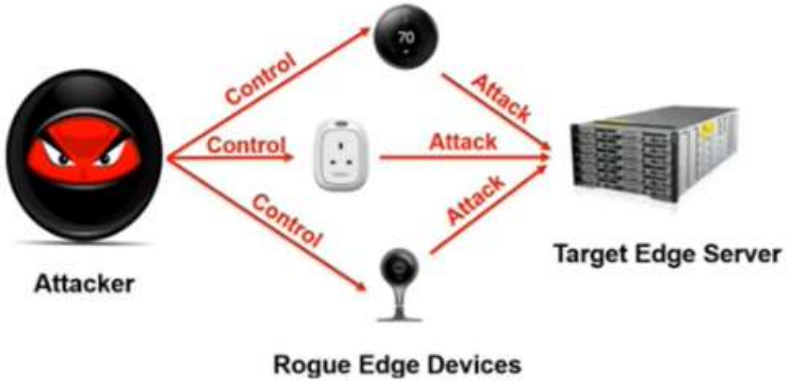
DDoS - type of cyberattack in which attackers aim to disrupt normal services provided by one or more servers based on distributed resources such as a cluster of compromised edge devices

Edge servers are more susceptible to DDoS attacks since they are relatively computationally less powerful to maintain strong defense systems as cloud servers do

DDoS attacks

DDoS attacks:

- Flooding-based attacks
- Zero-day DDoS attacks



Flooding-based attack technique

- UDP flooding
- ICMP flooding
- SYN flooding
- Ping of death (PoD)
- HTTP flooding
- Slowloris

Flooding-based Attack Mechanism

UDP flooding attack, an attacker continuously sends a large amount of ***noisy UDP packets*** to a target edge server

ICMP flooding attack, an attacker exploits the ICMP protocol to attack by sending a large number of ICMP Echo Request packets to a target edge server as ***fast*** as possible ***without waiting for the replies***. This type of attack consumes both outgoing and incoming throughputs of the victim server

SYN flooding attack, an attacker exploits the three-way handshake of the transmission control protocol (TCP) by initiating a huge amount of SYN requests with a spoofed IP address to a target edge server

Flooding-based Attack Mechanism

PoD attack, an attacker creates an IP packet with malformed or malicious content that has a length greatly larger than the maximum frame size for a standard IP packet (65 535 bytes) and splits the long IP packet into multiple fragments and sends them to a target server.

HTTP flooding attack, an attacker simply sends a large amount of HTTP GET, POST, PUT, or other legitimate requests to an edge server

Slowloris attack, an attacker creates numerous partial HTTP connections which can be realized by only sending HTTP headers to a target server but never completing one

Zero-day DDoS Attack

A zero-day DDOS attack

advanced than flooding-based DDoS

but it is more difficult to implement

attacker must find an unknown vulnerability (i.e., zero-day vulnerability) in a piece of code running on the target edge server/device

which can cause memory corruption and finally result in a service shutdown

Common vulnerabilities and exposures (CVE)- 2010-3972

Heap-based overflow that can cause a DoS on Internet Information Services (IIS) 7.0 and IIS 7.5

Defense Solution

- Root cause
  - Flooding-based attacks
    - Protocol-level design flaws/vulnerabilities within the network communication protocols
  - Zero-day attacks
    - Code-level vulnerabilities

Defense Solution against flooding-based attacks

Two categories:

- per-packet-based detection
- statistics-based detection

Per-packet-based detection:

- detecting and filtering malicious packets
- integrating packet filtering mechanisms into congestion control frameworks

Proposed ways in per-packet-based detection

spot DDoS on a per-packet basis having the same identifier

changing the identifiers of the packets using tools such as hping3

negative selection algorithm: whether the IP address of a packet is legitimated based on the eigenvalue sets to resist this type of DDoS

server to maintain a list of legitimate IP addresses

Proposed ways in Statistics-based detection

- Machine learning-based mechanisms to detect DDoS traffics
  - methods : J48, naïve Bayes, and Bayesian network classifiers to detect botnet DDoS
  - deep learning model using an autoencoder to detect encrypted DDoS traffics
  - neural networks to identify DDoS attacks in software-defined networks
- Benefit: little human effort
- But they may perform differently on different types of DDoS attacks

Defense solutions against zero-day attack

Mechanisms to spot possible memory leaks in a program :

- Pointer taintedness detection
- ECC-memory

But these methods require the original source codes, which are usually unavailable for edge devices

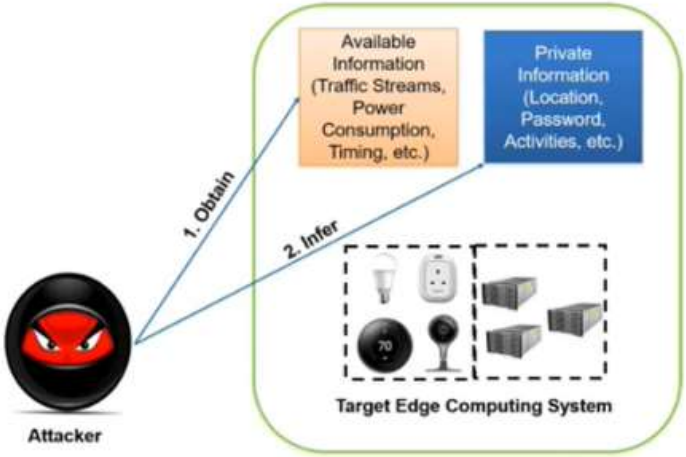
Deep learning models, e.g., recurrent neural networks (RNNs), graph neural networks (GNNs), and deep natural language processing (NLP), used to identify vulnerabilities in firmware with higher accuracy rates

But encryption and antidebug fuses are used in firmwares

An IoT firewall using software-defined networking (SDN) to reduce the attack surface of an exposed IoT device

Side-Channel Attacks and Defense Mechanisms

Definition: “A side-channel attack is **a security exploit that aims to gather information from or influence the program execution of a system** by measuring or exploiting indirect effects of the system or its hardware -- rather than targeting the program or its code directly.”



Attacks exploiting communication channel

Communication signals has a high potential to reveal sensitive information of a victim due to the rich channel information

Two types:

- exploiting packet streams
- exploiting wave signals

Exploiting packet stream

A packet is the atomic unit in most communication channels.

Different coding scheme employed by H.264 and MPEG-4 to reduce temporal redundancy in adjacent video frames can cause severe privacy leak in home surveillance

Simple machine learning algorithms such as k-nearest neighbors (k-NN) and density-based spatial clustering of applications with noise (DBSCAN) - accuracy of 95.8%

to infer - four standard human daily activities

dressing, styling hair, moving, and eating

Exploiting packet stream

Attacks can be launched by exploiting the IoT traffic streams

Attack mechanism: a three-step attack

- First separating the traffic into individual device flows using the IP addresses of the edge servers
- Correlating flow with its responsible IoT device according to unique identifiers
- Inferring user activities from the traffic rate changes

Exploiting wave signals

Using EMI (electromagnetic interference)

- Attack to infer the video content playing in modern TVs through the discernible EMI signatures
- Intentional EMI (IEMI), an attacker can manipulate the input and output signals of an IoT sensory device

Using Wi-Fi: as side channels

- A malware-less side-channel attack by exploiting channel state information (CSI) to infer a victim's sensitive password input such as Alipay code based on the finger movements

Exploiting wave signals

Brain–computer interfaces (BCIs) - attacker can successfully capture the raw electroencephalography (EEG) data (i.e., human brain wave data), and combine with machine learning algorithms

- Boosted logistic regression
- Stepwise linear discriminant analysis (SWLDA)
- Fisher's linear discriminant analysis (FLDA)

Attacker can infer victim's banking information, month of birth, face, and geographic location with the accuracies of 15%–40% better than the random guessing attack

Attacks exploiting power consumption

Power consumption is an indicator of the electric usage of a system

Information related to:

- the device that consumes the energy
- the intensity of computations in a computing task

Two types of attacks:

- exploiting power consumption collected by meters
- exploiting power consumption collected by oscilloscopes

Exploiting power consumption collected by meters

Smart meters can accurately measure the electric power consumption of a household

Side-channel inference method named nonintrusive appliance load monitoring (NILM) to monitor simple device states, e.g., ON or OFF, based on the energy consumption of individual appliances

Revised NILM - inference attack to show household activities, such as cooking, washing, laundering, watching TV, gaming

inferred from the energy data available in a smart meter infrastructure

Exploiting power consumption collected by oscilloscopes

- Oscilloscope is an instrument measuring the electronic information (e.g., voltage and current) of a hardware device
- In embedded devices: chip can perform cryptographic algorithms such as AES-CCM, with a hardcoded secret key in the chip
- Research has found that the power consumption of the hardware may be susceptible to leaking the key
- Adopting correlation power analysis can completely reverse the AES-CCM master key used to encrypt/decrypt the firmware installed in the Philips hue smart lights to create any malicious firmware and install on any Philips hue smart light over-the-air

Attacks exploiting smartphone-based channels

Smartphones have
- advanced OS
- Rich system information

Two types of attacks:
- exploiting the /proc filesystem
- exploiting the smartphone embedded sensors

Exploiting the /proc filesystem

/proc is a system-level filesystem created by the kernel in Linux – used for side-channel attacks

Contains the system information such as interrupt and network data

It is readable by the user-level threads and applications

Exploiting the smartphone embedded sensors

- Infer a user's keystroke by analyzing the acoustic sounds emitted from the physical keyboards
- Cracked the pattern lock of a smartphone by leveraging the acoustic signals reflected by the fingertip captured through microphones
- Tap keystrokes can be inferred using the smartphone accelerometer and gyroscope sensors
- Victim's eye movements from a video secretly recorded by a smartphone camera, to infer the victim's keystrokes on a mobile device

Defense Solutions

Defenses against side-channel attacks can be performed from two directions:

- restricting the accesses to side-channel information
- protecting the sensitive data from inference attacks

no feasible defense mechanism that can restrict the access to uncontrollable side channels

well-researched technique for protecting sensitive data from inference attacks - Data perturbation

Malware Injection Attacks and Defense Mechanism

- The action to effectively and stealthily inject/install malware into a computing system is called malware injection attack.
- This type of attacks is one of the most dangerous ones malware is a significant threat to system security and data integrity.
- Cloud has strong computational power to support high-performance firewall or other threat protection systems
- But edge servers and devices are vulnerable to malware injection attacks

Malware injection attacks

Malware injection attacks in edge computing into two categories:

- Server-side injections (injection attacks targeting edge servers)
- Device-side injections (injection attacks targeting edge devices)

Server-side injections

There are mainly four types of injection attacks targeting edge servers

- SQL injection
- Cross-site scripting (XSS)
- Cross-Site Request Forgery (CSRF)
- Server-Side Request Forgery (SSRF)

Device-side injections

The most common approach to remotely inject malware is to exploit the zero-day vulnerabilities that can lead to remote code execution (RCE)

- "IoT Reaper" virus captured in 2017
  - infected millions of IoT devices through the Internet protocol and Wi-Fi by exploiting at least 30 RCE vulnerabilities existing in 9 different IoT devices ranging from the network router to IP camera
- Smart Nest Thermostat lacks proper protection for firmware update, allowing an attacker to update an arbitrary firmware using a USB connection