Building Trustworthy IoT Data Lifecycle

Surya Nepal
Group Leader, Distributed Systems Security

www.data61.csiro.au
AUSTRALIA’S DIGITAL INNOVATION POWERHOUSE

1100+ employees [including students]
415+ students
31 Government partners
91 Corporate partners

29 University partners
190+ data-driven projects
172 patents
Three Challenges

Collectively defined with our defence partner, DST Group

- Building trustworthy and resilient cyber systems.
- Risk-based cyber approaches and shared awareness.
- Strengthening the human and social dimension of cyber security.
Focus areas within Data61 and in our D61+ network

- Trustworthy Systems
- Automating Cybersecurity and Resilient Systems
- Cyber-Physical Systems Security
- Quantitative Cybersecurity Risk Management
- Data Security and Privacy
- Data and Decision Trustworthiness
- Usable Human-centric Security
IoT Data Lifecycle
Internet of Things

• No single universally accepted definition.
  • Researchers have defined their own way.

• NIST builds Network of Things (NoT) model to define IoT

• NoT Model
  • Four fundamentals
    – Sensing
    – Computing
    – Communication
    – Actuation
  • Five building blocks
Data to Decision Life Cycle in IoT

CREATE → COMMUNICATE → AGGREGATE → ANALYZE → ACT

**Sensors**
- Physical devices
  - Sensor devices

**Network**
- Connectivity
  - Processing elements
  - Connectivity elements
  - Connectivity frameworks

**Integration**
- Management
  - IoT middleware
- Accumulation
  - Data ingestion
  - Data storage
  - Data abstraction
- Existing data
  - Unstructured data
  - Structured data

**Augmented intelligence**
- Processing
  - Processing engines/frameworks
  - Data messaging
- Analytics
  - Stream analytics (data in motion)
  - Batch analytics, machine learning (data at rest)

**Augmented behavior**
- Applications
  - User applications
  - Reporting
  - Edge computing
    - Intelligent gateways
    - Fog computing platforms

**Standards**

**Security**

Source: Deloitte’s IoT Reference Architecture.

Graphic: Deloitte University Press | DUPress.com
IoT Security Risk

The model of the NoT includes six elements—environment, cost, geographic location, owner, snapshot-in-time and a unique device ID—that all play a role in the reliability and security of a NoT.

IoT Triad
- Authentication
- Integrity
- Confidentiality

Things to Protect
- Device
- Communication
- Computation
- Decision
- Actuator

Distributed Computing
Security Challenges

• Zero Trust Architecture
  – “never trust, always verify” principle.
  – Zero Trust allows for no default trust for any entity (users, devices, applications, packets, and so on) regardless of its type or whether it’s on or related to the corporate network.
  – Blockchain?

• Deperimeterization
  – “multi-level protection” principle
  – A hardened perimeter security strategy is impossible to sustain within an agile business model.
  – Deperimeterization protects user data on multiple levels using encryption and dynamic data level authentication

• Software Defined Perimeters
  – “a cryptographic perimeter” principle
  – Only authenticated and authorized data are allowed to pass through.

Security and Computational Complexity

• Cryptographic Systems
  – Computationally infeasible; takes far too many resources to actually compute;
  – an infeasible computation’s cost is greater than the reward obtained by computing it;

• Distributed Ledger
  – Distributed ledgers reside on a distributed network of computers that are incentivized to collaborate on the maintenance of the ledger;
  – Consensus protocols;
Perfect Storm?

Computing devices are getting smaller and low-powered

Computing devices are becoming more powerful

Post Quantum Cryptography Algorithms for IoT!
IoT Security

IoT security is challenging because of

- Heterogeneous Networks
- Limited Resources
- Un-patchable things
- Untrusted environment
- Lack of well-defined perimeter
- Difficulty in Identity management
- New Firmware
- ....

Which bring the research challenges on

- **Platform**: Secure and modular hardware and software components
- **Technology**: Authentication of devices
- **Architecture**: Secure Networking/Software Architecture
Software Defined Perimeters

Kallol Krishna Karmakar  KallolKrishna.Karmakar@uon.edu.au
Vijay Varadharajan  Vijay.Varadharajan@newcastle.edu.au
Nepal, Surya  Surya.Nepal@data61.csiro.au
What is SDN?

- Software Defined Networking (SDN) is a centralized software centric control over forwarding hardware. Control and forwarding planes are decoupled in SDN for flexibility.

- Software Defined Networking (SDN) is an emerging architecture that is dynamic, manageable, cost-effective, and adaptable, making it ideal for the high-bandwidth, dynamic nature of today's applications.
Normal Operation of SDN:
- In NorthBound apps are running, and SDN controller core is the main interface which coordinates the whole operation of the SouthBound.
- First packet comes from the host machine, OpenFlow switch has no flow entry so it sends a packet_In request to core.
- Core takes decision according to the forwarding app and installs flows to switch to guide the packet.
## IoT Architecture

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
</table>
| Perception Layer | • An interface to the physical world and consists of actuators and sensors.  
                  • Transfers the raw data to the Network Layer |
| Network Layer  | • Processes and routes the data through the network infrastructure |
| Middleware Layer | • Further processing of the data happens |
| Application Layer | • Third-party businesses to develop and execute their applications for storage and further processing of the device data. |
Problem Domain

- Due to resource constrained nature of IoT devices, it is hard to integrate security measure into devices.

- Due to position, low energy and security measures a device can be turned into a Zombie. Thus, a single IoT devices can compromise the whole network infrastructure.

- Need a secure IoT network infrastructure.
# IoT Threats

<table>
<thead>
<tr>
<th>Layers</th>
<th>Components</th>
<th>Tasks</th>
<th>Threats</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>Third-party Applications, Websites, Consoles, Touch panel, etc.</td>
<td>Machine Learning, Business Models, Graphs, Flow Charts, etc.</td>
<td>(e) Malware/ Cross-site Scripting Issues, (f) Data Privacy Issues, (f) Buffer Overflow Attack</td>
</tr>
<tr>
<td>Middleware Layer</td>
<td>Vendor Specific Third-party Application</td>
<td>Machine Learning, Processing, Pre-processing, Real-time Action, etc.</td>
<td></td>
</tr>
<tr>
<td>Network Layer</td>
<td>Nodes, Gateways, Firmwares</td>
<td>Device Management, Processing and Secure Routing</td>
<td>(a) Spoofing, (b) Distributed Denial of Service (c) Man-in-The-Middle (d) Replay</td>
</tr>
<tr>
<td>Perception Layer</td>
<td>Sensors (Temperature, Humidity etc.) and Actuators (Motors, Relays etc.)</td>
<td>Identify, Monitor, Acquisition, Action, etc.</td>
<td></td>
</tr>
</tbody>
</table>
Our Contribution

- A SDN based security architecture that uses a policy based approach to secure IoT network infrastructure and detect malicious IoT devices and attacks.
- Authentication of IoT devices using a light weight protocol, which is in turn used in the provisioning of network services to authenticated IoT devices.
- Authorized access to network services by authenticated devices using secure OAuth protocol.
- Demonstration of the proposed security architecture and protocols using some realistic IoT scenario, showing how it can protect IoT infrastructure from attacks such as DDoS (Mirai) and Man-in-the-Middle (MITM) attack.
SDN IoT architecture

Developed Security Applications
- Enforcer
- Evaluation Engine
- Repositories
- Policy Manager

Other Network Applications
- IoT Authority
- IoT Authenticator
- IoT Security Application

NorthBound Interface
SDN Core Applications and Modules

SouthBound Interface (Forwarding Devices)
OpenFlow Switches
SW1, SW2, SW3, SW4

IoT Gateways
GW1, GWII

IoT Nodes
NI, NII

Sensors/Actuators

*GW: IoT Gateway
*N: IoT Node
*PbSA: Policy-based Security Application
SDN IoT Operation

1. Device
   - Calculates: $a \in Z_p^*$
   - Generates $T_d = aP$

2. Gateways/OF AP
   - S0: X = ID + Hello

3. IoT Authority
   - S1: Packet_IN(X)
   - S3: Initiate ECC device authentication procedure
   - S4: Send $(T_d, ID_d, R_d)$

4. IoT Authenticator
   - Calculates: $b \in Z_p^*$
   - $T_s = b + y_{gt}$
   - $T_{gt} = T_s$
   - $k_{gt=d} = T_s(R_d + H2(ID_d, y_{gt})P_{pub} + T_d)$
   - $M1 = H1(0, k_{gt=d})$

5. PbSA
   - Checks Policy Expression
     - If ok then provides Authorization(P)

6. Device
   - Generates OAuth Token (Ko)

7. IoT Authenticator
   - S8: Authorization Permission(Kp)

ECC procedure

OAuth procedure
Attributes:
- Flow based attributes: Flow ID, Type of Packets etc.
- Device attributes: Device ID
- Autonomous System Domain Attributes: AS ID, AS type, Src/ Des AS, etc.
- Switches Attributes: Security Label
- Host Attributes: Src/Des IP, etc.
- Flow/ Domain Constraints
- Path

Syntax:
\[ PE_i = <FlowID, IoT Device ID, SrcAS, DesAS, SrcHostIP, DesHostIP, SrcMAC, DesMac, User, FlowCons, DomCons, Services, Sec \rightarrow Profile, Seq \rightarrow Path >: <Actions > \]
How Mirai Works?
- It generates UDP Flood, SYN Flood, ACK Flood, TCP Stomp Flood, DNS Flood, HTTP Flood, GRE IP Flood, and GRE Ethernet Flood

- Used Telnet blocking policies.
- Even though the IoT device is physically compromised, they will not be able to infect others.
Performances

Policy Vs Throughput
(Four OpenFlow Switches)

Average Throughput (flows/ms)

<table>
<thead>
<tr>
<th>No of Policy Expression</th>
<th>20 POLICY</th>
<th>30 POLICY</th>
<th>40 POLICY</th>
<th>50 POLICY</th>
<th>WITHOUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5171.19</td>
<td>4329.93</td>
<td>2575.01</td>
<td>2510.74</td>
<td>15863.84</td>
</tr>
</tbody>
</table>

No of Policy Expression
Performances
Secure and Modular IoT Platform
Push the Policies to the Edge to Increase Resiliency
Mashup App
Our Solution: Levelled Authentication

• A new lightweight public key encryption scheme
  – Used to implement levelled authentication

• Feature
  – Small ciphertexts for a small message space at a required security level
  – Each authentication level exchanges a small encrypted nonce.
  – For example, at the 128-bit security level, our scheme generates
    – 22-byte ciphertexts for 2-byte messages
Internet of Things (IoT) Security Framework

Improve IoT security capabilities

Data61’s DTLS solution for IoT devices

• Lattice-based lightweight encryption and authentication schemes
• Multi-level and graded security characteristics for security differentiation
• Integrated into DTLS stack

Reduce IoT deployment complexity

Data61’s IoT device mashup platform

• Easy deployment and configuration
• Execute conditional actions
• Enforce Security Policies
Some Other Challenges
Asymmetric Resources

How do you take advantage of these asymmetric resources to find the most efficient and effective security and privacy solution?

<table>
<thead>
<tr>
<th>Layer</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLICATION</td>
<td>CoAP</td>
</tr>
<tr>
<td>TRANSPORT</td>
<td>UDP</td>
</tr>
<tr>
<td>NETWORK</td>
<td>IPv6/RPL</td>
</tr>
<tr>
<td>ADAPTATION</td>
<td>6LoWPAN Adaptation</td>
</tr>
<tr>
<td>MAC</td>
<td>IEEE 802.15.4</td>
</tr>
<tr>
<td>PHYSICAL</td>
<td>IEEE 802.15.4</td>
</tr>
</tbody>
</table>

Cloud – Elastic Resources
Edge – Finite Resources
IOT – Limited Resources

CLIENT SERVER

PEER TO PEER
Security Foraging

Self* Access Policies

• Recalling ...
  – the NoT includes six elements—
    – environment, cost, geographic location, owner, snapshot-in-time and a unique device ID—
    – that all play a role in the reliability;

• Self Generation Access Control Policies
  – access control needs to become an integral part of the resources being protected;
  – access control needs to be generated dynamically by the resources depending on the context in which they are being used.
Building Secure Systems

- Donald Rumsfeld (2002)
  - Known knowns
  - Known unknowns
  - Unknown unknowns
- Left out
  - Unknown knowns

- Security is about modelling these threats through science and building engineering solutions
  - In science, the value of a security model lies on how well it matches with the behaviours of the physical world (can I make a security model of this thing?); \(\rightarrow\) decrease the model;
  - In engineering, the value of the secure physical system lies on how well it matches with the behaviour of the model (can I make a secure thing out of this model?); \(\rightarrow\) increase the models;
Thanks!