Northrop Grumman Cybersecurity Research Consortium (NGCRC) 2017 Fall Symposium

Secure / Resilient Systems and Data Dissemination / Provenance

26 October 2017
Bharat Bhargava
Purdue University

Technical Champion(s): Donald Steiner, Paul Conoval, Daniel Goodwin, Jason Kobes, Kory Brin
Data Dissemination/Provenance
Collaboration with NGC and MIT:

- "WAXEDPRUNE" project: Web-based Access to Encrypted Data Processing in Untrusted Environments

Researchers:

- Donald Steiner, Jason Kobes, Leon Li, Paul Conoval (Northrop Grumman Corporation)
- Bharat Bhargava, Denis Ulybyshev, Miguel Villarreal-Vasquez, Aala Alsalem, Leszek Lilien, Ganapathy Mani, Mai Elkady, Balamurugan Anandan, Rohit Ranchal (Purdue University)
- Harry Halpin (MIT)
Outline

• Collaboration with NGC IRADs
• Problem Statement
• Research Approach
• Benefits
• Core Design
• Demos and Experiments
• Proposed Deliverables
• Ongoing Work
Collaboration with NGC IRADs

All materials available here: https://www.cs.purdue.edu/homes/bb/#research

- **WAXEDPRUNE project:**
  - Proposed by Donald Steiner, Leon Li, Jason Kobes
  - Extended prototype is running and can be accessed at [www.waxedprune.cs.purdue.edu](http://www.waxedprune.cs.purdue.edu)
  - Demo [https://www.dropbox.com/s/4wg3vv5j4s16v/NGCRC-2017-Bhargava-Demo1.wmv?dl=0](https://www.dropbox.com/s/4wg3vv5j4s16v/NGCRC-2017-Bhargava-Demo1.wmv?dl=0)
  - Code [https://github.com/Denis-Ulybysh/absoa17](https://github.com/Denis-Ulybysh/absoa17)

- **MTD project:**
  - Code [https://www.dropbox.com/s/frsflh0xbhewp4u/mayflies.py?dl=0](https://www.dropbox.com/s/frsflh0xbhewp4u/mayflies.py?dl=0)

**IEEE Cloud 2017 papers:**


D. Ulybyshhev, B. Bhargava, D. Steiner, L. Li, J. Kobes, A. Alsalem, H. Halpin and R. Ranchal
“Secure Data Sharing and Data Leakage Detection in Untrusted Cloud”
Awards

1. **Schleman Gold Medallion Award, 2017**
   For supporting women in computer science

2. **Focus Award at Purdue University, 2017**
   Applying research to help differently abled (visually impaired, blind)

3. **Purdue University Computer Science Corporate Partners Research Award 2017**
   For work based on WaxedPrune project extension
Expertise

• Data Privacy
• Data Provenance
• Insider Threat Detection
• Encrypted search over encrypted data
• Innovative use of blockchain for trust
• Moving Target Defense
• Applied Machine Learning
• Cognitive Autonomy
• Knowledge Discovery
• Scalability in SDN
• Experience of building prototypes for NG TechFest
Focus: Data Privacy, Provenance, Data Leakage

- Detect data leakage (intentional or inadvertent) to unauthorized services
- Track provenance to support data lineage, reproducibility
- Measure data leakage (what got leaked, when, to whom, how sensitive was the data)
- Support encrypted search over database of Active Bundles
Problem Statement

Cloud Server

Data Owner

Active Bundle

Patient’s EHR (Active Bundle)
Contact, Medical and Billing Information

Doctor
- Contact Info
- Medical Info
- Billing Info

Web Crypto Authentication

Doctor
- Contact Info
- Medical Info
- Billing Info

Web Crypto Authentication

Insurance
- Contact Info
- E(Medical Info)
- Billing Info

Web Crypto Authentication

Researcher
- E(Contact Info)
- Medical Info
- Billing Info

Web Crypto Authentication

Leakage of Medical Info

Scenario of EHR Dissemination in Cloud and Leakage (suggested by Dr. Leon Li, NGC)
Problem Statement

Search Index (CryptDB)

Web-based Application

Search by keywords

Retrive and Display Documents

Public Cloud

Active Bundle Database (Hospital IS)

Build search index

- \( AB_1 \)
- \( AB_2 \)
- \( \ldots \)
- \( AB_n \)

Encryptd Search over Encrypted Data stored in Cloud (suggested by Dr. Leon Li, NGC)

Example:

```
SELECT prescription FROM Hospital_IS WHERE diagnosis = 'insomnia';
```
Privacy-Preserving Data Dissemination based on:

- Access control policies
- Trust level of a subject (service, user)
- Context (e.g. emergency vs. normal)
- Security level of client’s browser (crypto capabilities)
- Authentication method (password-based, fingerprint, etc)
- Source network (secure intranet vs. unknown network)
- Type of client’s device: desktop vs. mobile (detected by Authentication Server)
Data Leakage Detection

- For leaked encrypted data:
  - Based on Obligations: how data is used by authorized party
  - Obligations are enforced by Central Monitor when recipient tries to extract data from Active Bundle (AB)
  - Digital watermarks that can be checked by web crawlers
- For leaked decrypted data:
  - Based on visual watermarks embedded into sensitive data
WAXEDPRUNE prototype is extended to support the following features:

1. Data leakage detection in SOA
2. Encrypted search over encrypted data (over Active Bundle database)

Prototype is implemented and was demonstrated at NGCRC symposium in April 2017

- Source code and Demo video [7] are available
Novelty

Contributes to Data Privacy, Integrity and Confidentiality

• On-the-fly key generation scheme
• Does not require data owner’s availability
• Supports on-the-fly data updates for multiple subjects
• Tamper-resistance: data and policies integrity is provided
• Supports encrypted search over database of Active Bundles
• Data leakage detection and leakage damage assessment
• Captures data provenance for use in leakage measure and forensics
Active Bundle (AB) [2,4]: contains

- **Sensitive data:**
  - Encrypted data items

- **Metadata:** describe AB and its access control policies
  - Policies manage AB interaction with services and hosts

- **Policy Engine:** enforces policies specified in AB
  - Provides tamper-resistance of AB [4, 1]
Data Provenance

- CM monitors each decryption and stores provenance data
  - Who tried to decrypt data
  - Where did data come from
  - What type (class) of data
  - When

- Provenance data can be corrupted
  - Send provenance messages to CM via secure protocol (https)
  - Use blockchain in the future to provide provenance data integrity

- Provenance data itself can be leaked
  - Encrypt it, key is stored at trusted party

- Provenance data is used to investigate data leakage
How can data get leaked by authorized subject?

- In the form of encrypted data (the whole AB is leaked):
  - Data is protected by AB, but leakage needs to be detected
  - Detection Phase 1: digital watermark can be checked by web crawler to detect copyright violations
  - Detection Phase 2: based on Obligations: how data is used by authorized party?
    - Obligations are enforced by Central Monitor (TTP) when recipient tries to extract data from AB
    - CM checks whether data is supposed to be where they are

Based on Discussions with Jason Kobes (NGC) at NGC Symposium 2016
Data Leakage Detection

AB contains:
- $\text{Enc} \ [\text{Data}(D)] = \{\text{Enc}_{k_1}(d_1), \ldots, \text{Enc}_{k_n}(d_n)\}$
- Access Control Policies ($P$) = $\{p_1, \ldots, p_k\}$

- Service $X$ is authorized to read $d_1$ from $AB$
- Service $X$ may leak decrypted $d_1$ or entire $AB$ to $Y$
- When $Y$ tries to decrypt $d_1$, $AB$ notifies $CM$
- $CM$ enforces obligation policies

* This work is used in PhD Thesis Proposal of Denis Ulybyshev, Purdue University
Data Leakage Detection

Data can get leaked by authorized subject in the form of:

- Decrypted (raw) data (picture taken on a smartphone camera):
  - Data is not protected by AB anymore
  - Detection based on visual watermarks embedded into data and steganography (e.g. word order encodes valuable info)

- Decrypted plaintext with watermark removed
  - Can give part of the data only to the requesting service and watch its trust level first
Data Leakage Mitigation Methods

- **Layered Approach**: Don't give all the data to the requester at once
  - First give part of data (incomplete, less sensitive)
  - Watch how it is used and monitor trust level of using service
  - If trust level is sufficient – give next portion of data

- **Raise the level of data classification** to prevent leakage repetition

- **Intentional leakage** to create uncertainty and lower data value

- **Use provenance data stored at CM** to identify the list of suspects

- **Monitor network messages**
  - Check whether they contain e.g. credit card number that satisfies specific pattern and can be validated using regular expressions [4]
Data Leakage Damage Assessment

- After data leakage is detected, damage is assessed based on:
  - To whom was the data leaked (unknown service with low trust level vs. service with high level of trust)
  - Sensitivity (Classification) of leaked data (classified vs. unclassified)
  - When was leaked data received (recent or old data)
  - Can other sensitive data be derived from the leaked data (i.e., diagnosis can be derived from leaked medical prescriptions)

\[
\text{Damage} = K_{\text{Data is Sensitive}} \times K_{\text{Service is Malicious}} \times S(t)
\]

, where \( S(t) \) is the function for data sensitivity in time
Data Leakage Timing

- Data-related event (e.g. final exam) occurs at $t_0$
- Threat from data being leaked before $t_0$ is high
- Threat from data being leaked after $t_0$:
  1) No threat at all
  2) Linearly decreases with time
  3) Remains constant (for highly-sensitive data)
Performance overhead (61%) with data leakage detection on / off
Anti-fragility

- After leakage is detected, make system stronger against other related attacks on privacy
  - Separate compromised role into two: Role and Benign_role
  - Send new certificates to all benign services with Benign_role
  - Create new Active Bundle with new policies, restricting access to Role (e.g. to all doctors from the same hospital with a malicious one)
  - Increase sensitivity level for leaked data items, i.e. for diagnosis
  - Disable “Save As” functionality on local node or exclude highly sensitive data from what can be stored locally
Encrypted Search over Encrypted Data

- Cloud provider hosts database of ABs
- AB contains data in encrypted form
- AB has extra-attribute(s) used for indexing (age, diagnosis)

Query example:

```
SELECT ID FROM EHR_DB WHERE age BETWEEN 35 AND 40;
```

Converted query:

```
SELECT c1 FROM Alias1
WHERE ESRCH ( Enc(age), Enc(35, 40) );
```

Result: {001, 003}

<table>
<thead>
<tr>
<th>ID</th>
<th>diagnosis</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>Insomnia</td>
<td>35</td>
</tr>
<tr>
<td>002</td>
<td>Cold-sore HSV1</td>
<td>30</td>
</tr>
<tr>
<td>003</td>
<td>Concussion</td>
<td>40</td>
</tr>
</tbody>
</table>
Deliverables

- **Prototype implementation:**
  - Data Leakage prototype
  - Active Bundle Module
    - AB implementation as an executable JAR file
    - AB API implementation using Apache Thrift RPC framework


- **Documentation:**
  - Deployment and user manual
  - Demo video “Data dissemination and leakage detection”

Current work: BlockHub

Blockchain-based Software Sharing (WaxedPrune extension)

- Blockchain-based technology to provide integrity of provenance data
- Framework for secure cross-domain software development
Current work: BlockHub

BlockHub can be used for the following:

1. Tracking and control of software components that are shared across multiple countries or security domains.
2. Automating the export auditing and tracking processes.
3. Cross-domain dissemination of encrypted software modules based on role-based and attribute-based access control.
4. Licensing provenance of deployed software components.
5. Enabling software supply chain that is tamper resistant.
Current work: BlockHub

⇒ X and Y can share software via *smart contracts* running in blockchain network
⇒ Every request and is logged in the blockchain’s distributed ledger
⇒ Software is transferred in case when authorization has been granted by both smart contract and policy enforcement engine of the AB (SB)


Secure/Resilient Systems
A Moving Target Defense Solution
Collaboration with NGC IRADs and Publications

Internal Research and Development:

• Cyber Resilient Systems (Daniel Goodwin)
• Enterprise Resiliency (Frank Wilson)
• Cloud-based Cyber Resiliency (Carlos Otero)

MTD Publications:


Collaboration with NGC IRADs

• **Cyber Resilient Systems IRAD**¹:
  - Cyber resiliency is based on the ability of the system to start secure, stay secure and return secure

- The system starts with trusted components
- Continue to operate maintaining level of trust
- Return to trusted state case of an event

¹NGC Cyber Resilient Systems IRAD (Daniel Goodwin)
Collaboration with NGC IRADs

• **Enterprise Resiliency IRAD**
  
  – Monitoring a family of systems and performing real-time analytics to provide actionable artifacts that can automatically address system anomalies
  
  – Identifies agnostics technologies and architecture patterns that facilitate solutions to maximize computation resources for the enterprise
  
  – Enterprise Anomaly Discovery
  
  – Enterprise Automatic Healing
Collaboration with NGC IRADs

- **Cloud-based Cyber Resiliency IRAD**:
  - Static and dynamic analysis of cloud applications
  - **Ensuring integrity during execution**
Outline

- Problem Statement
- Research Approach
- Benefits of Proposed Research
- MTD Approach Details
- MTD Architecture
- Prototype and Measurements (2016-2017 Final Report)
  - Methodology
  - Results
- Current Work
Focus: Secure/Resilient Systems

Attack Surface
Problem Statement

Focus: Secure/Resilient Systems

Replication approaches are used to improve resiliency.
Focus: Secure/Resilient Systems

Replication approaches in cloud computing increase the attack surface.
Problem Statement

Focus: Secure/Resilient Systems

We need resilient/self-healing systems that can accurately detect anomalies and dynamically adapt themselves to keep performing mission-critical functions even under attacks and failures.
Focus: Secure/Resilient Systems

- **Research Question:** Is it possible to construct a generic attack-resilient framework for distributed cloud systems with a combination of dynamic network configuration and continuous replacement of virtual machines?
Resiliency Solution = Live Monitoring + MTD
• “Stay one-step ahead” of sophisticated attack
  – Protect the entire stack through dynamic interval-based spatial randomization
  – Avoid threats in-time intervals rather than defending the entire runtime of systems through Mobility and Direction
  – System will start secure, stay secure and return secure
  – Increase agility, anti-fragility and adaptability of the system
  – Unified generic MTD framework that enables reasoning about behavior of deployed systems on cloud platforms
Research Approach

- **Adversaries have an asymmetric advantage:** They have the time to study a system, identify its vulnerabilities, and choose the time and place of attack to gain the maximum benefit.

- **The idea of moving-target defense (MTD):** Imposing the same asymmetric disadvantage on attackers by making systems dynamic and therefore harder to explore and predict.
Benefits of the Proposed Solution

• State of the Art System View:

At a given time only some layers of the stack (Application, OS or Network) are checked/protected.

Sate Verification

Time Intervals (< 1 sec)
Benefits of the Proposed Solution

- Proposed Solution System View:

At a given time all layers of the stack (Application, OS or Network) are checked/protected.
MTD Approach Details

- Nodes run a distributed application on a given platform for a controlled period of time.
- The running time is chosen in a way that successful ongoing attacks become ineffective.
- The new fresh machine will integrate to the system and continue running the application after its data is updated.

SDN Network
MTD Approach Details

- Nodes run a distributed application on a given platform for a controlled period of time.
- The running time is chosen in a way that successful ongoing attacks become ineffective.
- The new fresh machine will integrate to the system and continue running the application after its data is updated.

SDN Network
MTD Architecture

Components:
(1) Virtual Reincarnation (ViRA)   (3) SDN Network Dynamics
(2) Proactive Monitoring           (4) Systems States and Application Runtime
Virtual Reincarnation: ViRA

- Randomization and diversification technique where nodes (virtual machines) running a distributed application vanish and reappear on a different virtual state with different guest OS, Host OS, hypervisor, and hardware.
Virtual Reincarnation: ViRA

"Active machines are replaced by new ones with a totally new image"

SDN Network Dynamics

- Network devices are reconfigured via OpenFlow on-the-fly
- New added flows redirect traffic intended for the old machine to the new machine
**SDN Network Dynamics**

- Network devices are reconfigured via OpenFlow on-the-fly
- New added flows redirect traffic intended for the old machine to the new machine

OpenFlow Tables:
- `table=0,priority=0,actions=...`
- `table=1,priority=0,actions=...`
- `table=2,ip,nw_dst=10.0.0.10,...`
Prototype and Measurements

Implementation Details

- A Byzantine fault tolerant (BFT-SMaRt) distributed application was run on a set of Ubuntu (either 12.04 or 14.04 randomly selected).
- VMs are connected with an SDN network using Open vSwitch
- The reincarnation is stateless
- The set of new VMs are periodically refreshed to start clean
- The network is reconfigured using OpenFlow

OpenFlow Tables:
- table=0,priority=0,actions=...
- table=1,priority=0,actions=...
- table=2,ip,nw_dst=10.0.0.10,...

Host Machine: Linux 16.04 LTS
- Versions: 12.04, 14.04
1. **VM restart time**: Time it takes the machine to respond to be full operational since it is started.

2. **Virtual creation time**: Time to create the new image of the VM.

3. **Open vSwitch flow injection time**: Time it takes to inject new flows to Open vSwitch

<table>
<thead>
<tr>
<th>Measurements</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM restart time</td>
<td>~ 7s</td>
</tr>
<tr>
<td>VM creation time</td>
<td>~ 11s</td>
</tr>
<tr>
<td>Open vSwitch flow injection time</td>
<td>~ 250ms</td>
</tr>
</tbody>
</table>

**Note**: The important factor for system downtime here is the Open vSwitch flow injection time, as VM creation and restart take place before the reincarnation process.
Current Work

1. Current implementation

2. Live monitoring with AI techniques

3. From Stateless to Stateful Virtual Reincarnation
Current Work

**Stateful Virtual Reincarnation Support:**

- We preserve the state of the virtual machine during the reincarnation process to make the solution application-agnostic.
- Test the framework with Secure SOA Services (stateful reincarnation).
Stateful Reincarnation Ideas

- **D**: Synchronized Data
- **T**: Different version of Text
- **VM4** replaces **VM**

Current Work
Stateful Reincarnation Ideas

- Create different versions of binaries
- The original code is kept and set with read-only permission so that it can be used as part of the reference to the new locations of the blocks in the re-randomized version.
- We avoid identifying and updating code position pointers in each randomization process by keeping a table of trampolines as shown in (b). Each block is located at a fixed offset (i.e., off_c) with respect to the trampoline table.
- The pointers (in the original code space) are dynamically redirected to its respective address in the code variant when it is de-referenced.

Presentations and Publications

1. NGC Cyber Resilient Systems IRAD (http://www.northropgrumman.com)
2. Enterprise Resiliency IRAD (http://www.northropgrumman.com)
Backup Slides
Attribute- and Role-based Data Dissemination

- Recipients are authorized for different fragments of EHR
- EHR is AB, can be sent to Doctor, Insurance, Researcher

**AUTHENTICATED CLIENT**

- **Browser’s Crypto Level:** High
- **Authentication Method:** Fingerprint
- **Client’s device:** Desktop
- **Source network:** Corporate Intranet
- **Role:** Doctor

Q: What if Doctor leaks data to insurance agent for which the agent is not authorized?

**AUTHENTICATED CLIENT**

- **Browser’s Crypto Level:** Low
- **Authentication Method:** Password
- **Client’s device:** Mobile
- **Source network:** Unknown
- **Role:** Insurance Agent

*Discussed with Donald Steiner (NGC)*
AB Tamper – Resistance

- Key is not stored inside AB [2, 4]
- Separate symmetric key is used for each separate data value
- Protection against tampering attacks ensured

\[
\text{Aggregation}\{d_i\} \quad (\text{Execution info}; \quad \text{Digest(AB Modules)}; \quad \text{Resources})
\]

\[
\text{Aggregation}\{d_i\} \quad \text{(Tampered (Execution info}; \quad \text{Digest(AB Modules)}; \quad \text{Resources}))
\]

* Discussed with Jason Kobes (NGC)
Performance overhead of Active Bundle with detection of browser's crypto capabilities on / off