Northrop Grumman Cybersecurity Research Consortium (NGCRC)

Fall 2016 Kick-Off Meeting

Secure / Resilient Systems and Data Dissemination / Provenance

04 November 2016

Bharat Bhargava

Purdue University

Technical Champion(s): Donald Steiner, Jason Kobes, Leon Li, Sunil Lingayat, Daniel Goodwin, Frank Wilson

Problem Statement

Focus: Secure Data Dissemination / Provenance

- Authorized service can only access data items for which it is authorized
  - Encrypted Search over Active Bundles
- Detect data leakage to unauthorized services and report them to data owner
- Track provenance to support data lineage, reproducibility
- Measure data leakage (what got leaked, when, to where, how sensitive was the data)
- Experiments and Prototype

Outline

NORTHROP GRUMMAN

- Problem Statement
- Collaboration with NGC IRADs
- State of Current Technology
- Research Approach
- Benefits
- Core Design
- Demos and Experiments
- Proposed Deliverables
- Ongoing tasks

Collaboration with NGC

- "WaxedPrune" project: Web-based Access to Encrypted Data - Processing in Untrusted Environments
  - Preserves privacy and integrity of data
    - Donald Steiner
    - Leon Li
    - Jason Kobes
- CURATE2 project (Jason Kobes)

"Authentication of User's Device and Browser for Data Access in Untrusted Cloud," D. Ulybyshev, B. Bhargava, L. Li, J. Kobes, D. Steiner, H. Halpin, B. An, M. Villarreal, R. Ranchal. *CERIAS Security Symposium, April 2016*.

Paper submitted for ICDE-2017 Data Engineering conference:
D. Ulybyshev, B. Bhargava, L. Li, D. Steiner, J. Kobes, H. Halpin, M. Villarreal and R. Ranchal
"Privacy-Preserving Data Dissemination and Data Leakage Detection in SOA"

Examples of Recent Data Leakages

Company	Time	Incident Details
Adobe Systems	Oct.2013	150 million accounts of software subscription database got leaked
Anthem	Feb.2015	78.8 million of PII records got leaked
Experian Information Solutions and T-Mobile, USA	Sep.2015	Data (SSN, credit card information) of about 15 million customers who applied for credit got leaked
U.S. Office of Personnel Managenet: Agency of the U.S. Federal government	Jun.2015	SSN, names, addresses, places of birth of 22 million people got leaked

State of Current Technology

Policy-based Data Dissemination

- “Encore” (sticky policies) system is prone to Trusted Third Party (TTP)-related issues

Digital Rights Management (DRM) Systems

- Windows Media DRM (Microsoft)
  - Disseminates audio and video data over IP network
- “MediaSnap” (protects pdf documents)
- Hardware-based DRM
  - Resistant to security breaches in OS
  - Infeasible to change, bypass or uninstall security features

  Cons: higher costs, limited flexibility and less interoperability
- Watermarking
  - Digital (e.g. DCT for images), can be checked by web crawlers
  - Visual: supported by modern printers

Research Approach

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)

Research Approach (cont.)

Data Leakage Detection

- For leaked encrypted data:
    - Based on Obligations: how data is used by authorized party
    - Obligations are enforced by Central Monitor
- For leaked decrypted data:
    - Based on watermarks embedded into sensitive data
        - Digital watermarks
        - Visual watermarks

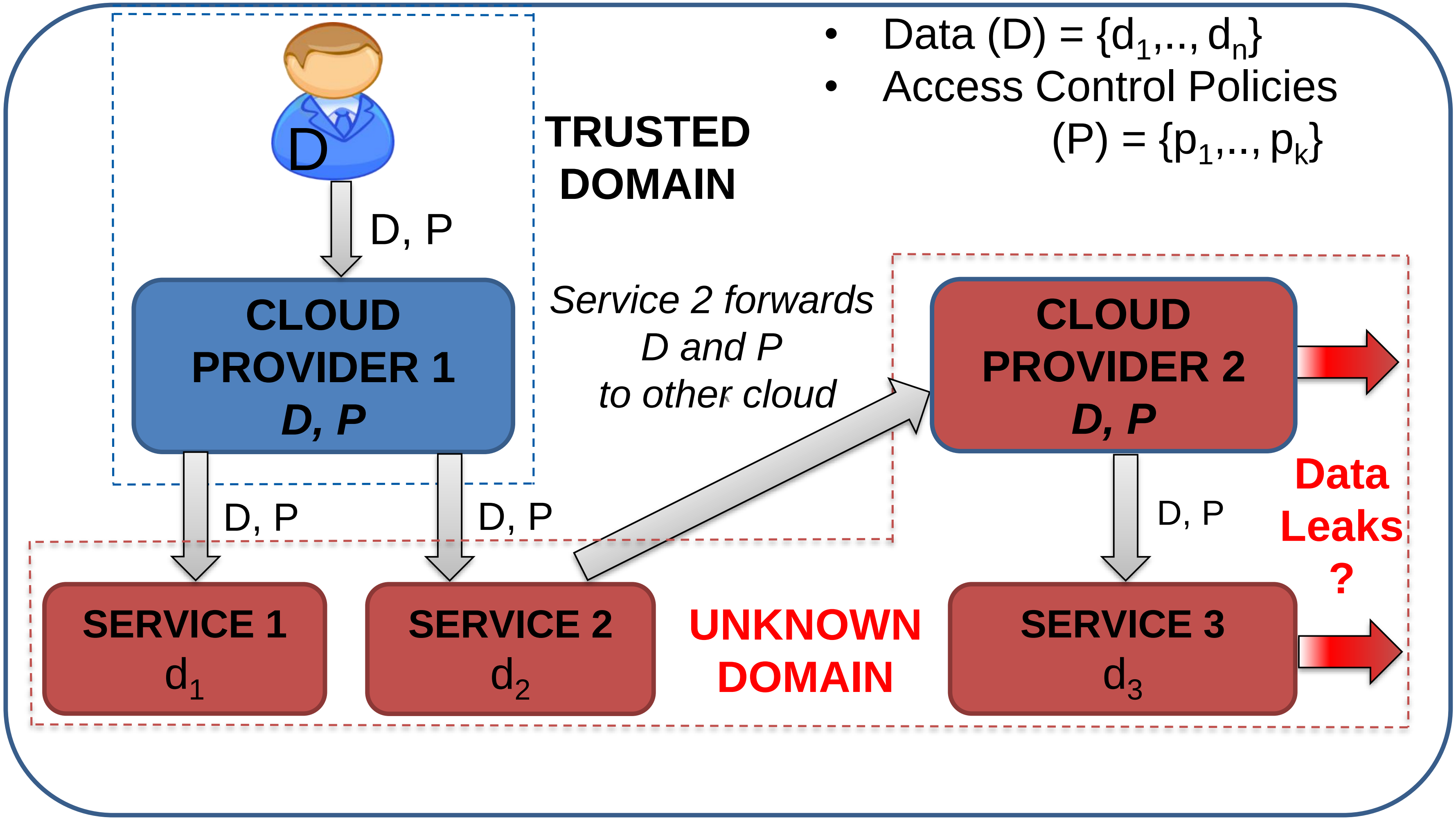
Benefits

Contributes to Data Privacy, Integrity and Confidentiality

- Does not require data owner’s availability
- Trust level of subjects is constantly recalculated
- Supports data and policy updates for multiple subjects
- Tamper-resistance: data and policies integrity is provided
- Supports encrypted search over database of ABs
- Data leakage detection and leakage damage assessment
- Captures data provenance for use in leakage measure and forensics
- Compatible with industry-standard SOA/cloud frameworks

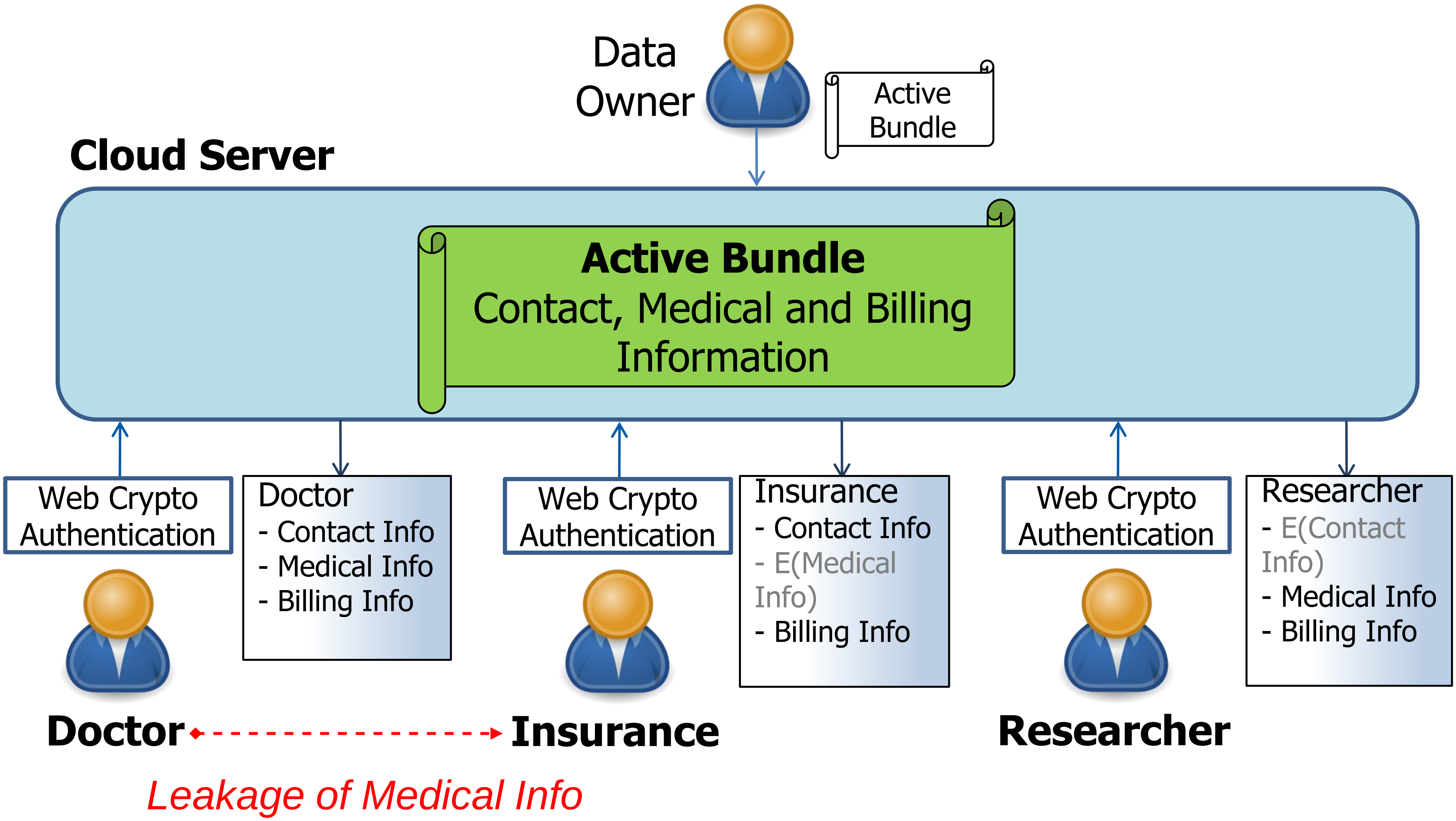
Data Leakage in Untrusted Cloud

- Data (D) = $\{d_1,.., d_n\}$
- Access Control Policies (P) = $\{p_1,.., p_k\}$



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

Extended Prototype for TechFest'17: EHR Dissemination with Data Leakage Detection



Data Owner

Active Bundle

Cloud Server

Active Bundle

Contact, Medical and Billing Information

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

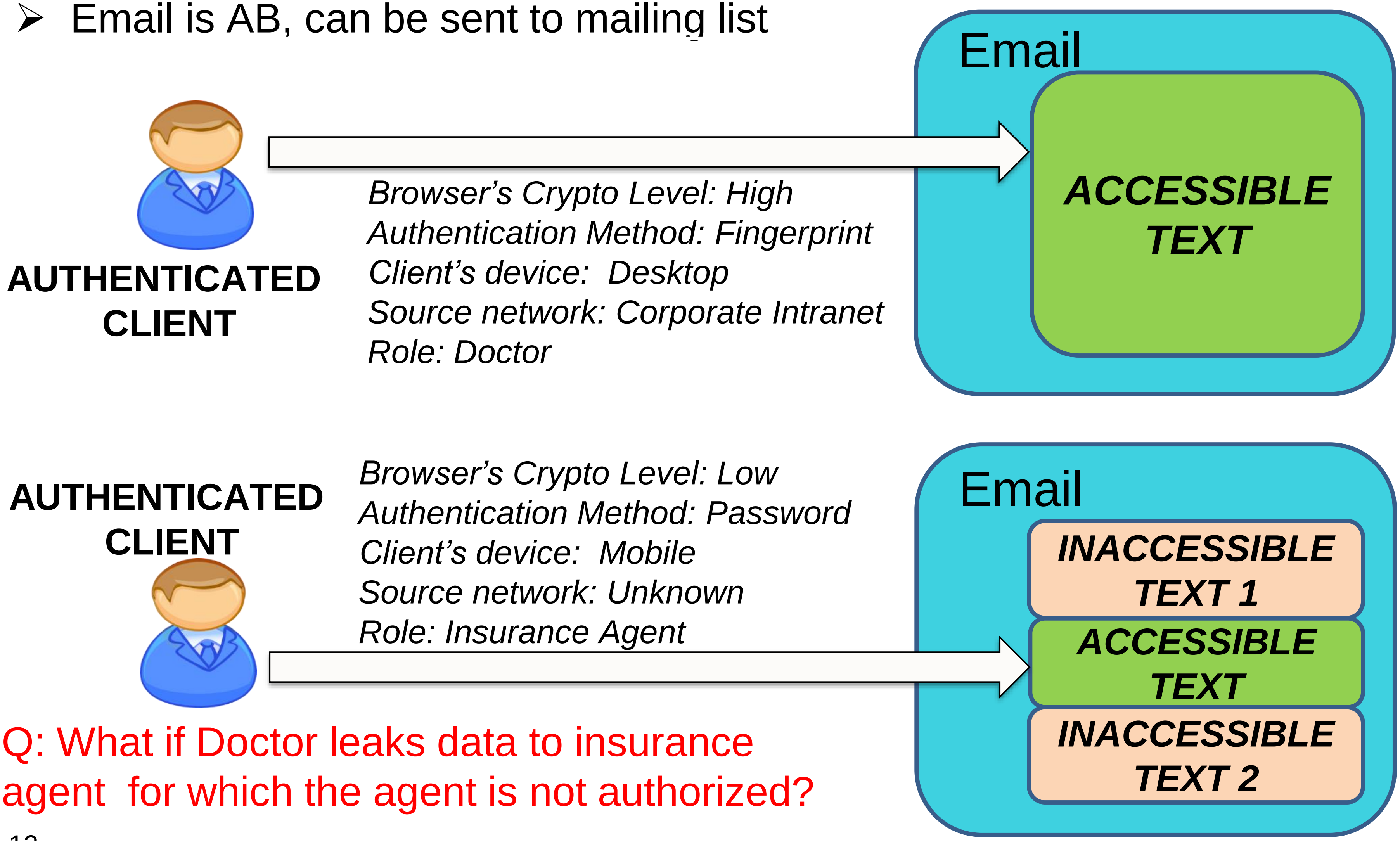
Doctor **Insurance** **Researcher**

Leakage of Medical Info

Scenario of EHR Dissemination in Cloud (by Dr. Leon Li, NGC)

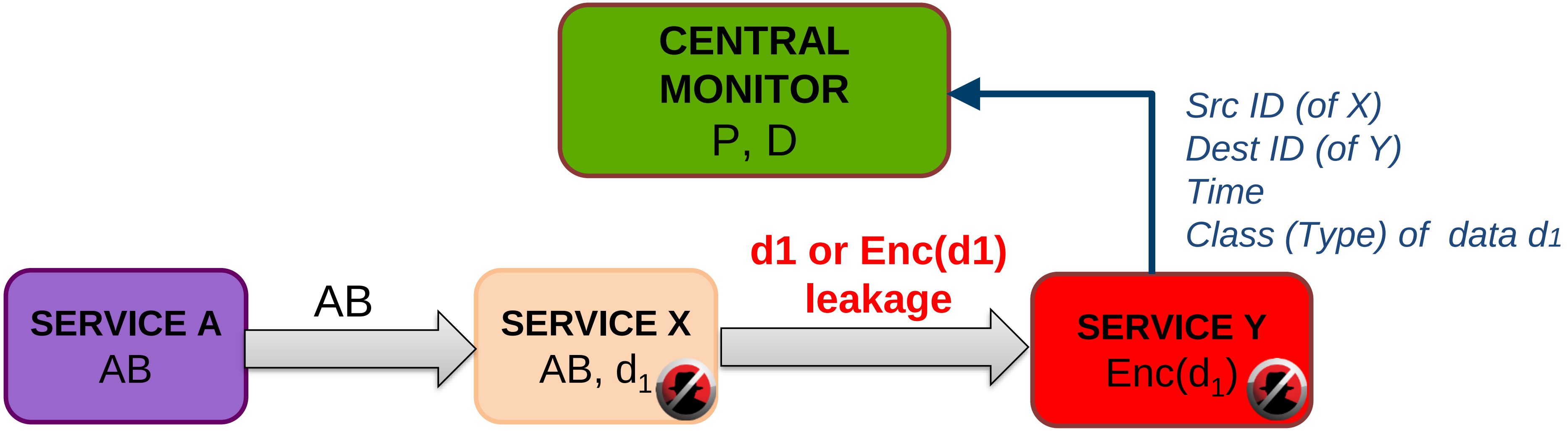
AB Use Case and Data Leakage

- Recipients are authorized for different fragments of email
- Email is AB, can be sent to mailing list



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

Core Design: Data Leakage Detection



AB contains:

- Enc [Data(D)] = {Enc_{k1} (d_1), ... , Enc_{kn} (d_n) }
- Access Control Policies (P) = {p_1,.., p_k}

- X is authorized to extract and read d_1 from AB
- X may leak plain d_1 or Enc(d_1) to Y

Types of Data Leakage

d_1 can be leaked as:

- Ciphertext
  - Data protected by AB
  - Leakage is detected by obligation enforcement
- Decrypted plaintext with watermark
  - Data not protected by AB anymore
    - Leakage can be detected by digital and visual watermarks
    - Watermarks can be used by web crawlers to detect copyright violations
- Decrypted plaintext without watermark
  - Data not protected, no leakage detection

Core Design: Data Leakage Detection

- Make data inseparable from AB and prevent plaintext leakage
  - Only our special software can decrypt and view data d_1 from AB
  - "Print Screen", "Save As" functionalities disabled
  - Additional activation on website is required
- Software notifies CM: d_1 arrived to Y from X
- CM checks against centralized Database of obligations: whether d_1 is supposed to be at Y. If NO then:
  - Blacklist X, Y
  - Reduce their trust level
  - Mark data d_1 as compromised and notify services about it

** Discussed at NGC Symposium (April, 2016) and during weekly NGC meetings with Jason Kobes, Leon Li, Donald Steiner, Paul Conoval*

Core Design: Type of Provenance Data

- CM monitors each transaction between services and stores provenance data
  - Who sent data
  - To Whom
  - What type (class) of data
  - When

- Data accesses / updates and policy updates are captured
- Provenance data can be corrupted
  - Send data provenance messages to CM via secure protocol (https)
  - Backup with trusted server
- Provenance data itself can be leaked
  - Solution: encrypt it, key is stored at trusted party
- Provenance data is used to investigate data leakage
  - *Example*: *find to whom did subject A send emails within last 10 days*

** Data leakage detection and provenance are part of NGC Proposal for 2016-2017*

Mitigation

- **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
  - If trust level is sufficient – give next portion of data
- **Use provenance data stored at CM** to identify the list of suspects
- **Raise the level of classification of data** to prevent leakage repetition
- **Intentional leakage** to create uncertainty and lower data value
- Monitor network messages
  - Check whether they contain e.g. credit card number which satisfies specific pattern and can be validated using regular expressions

Data Leakage Damage Assessment

- After data leakage is detected we assess damage 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 leaked 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)

Timing of Leaked Data

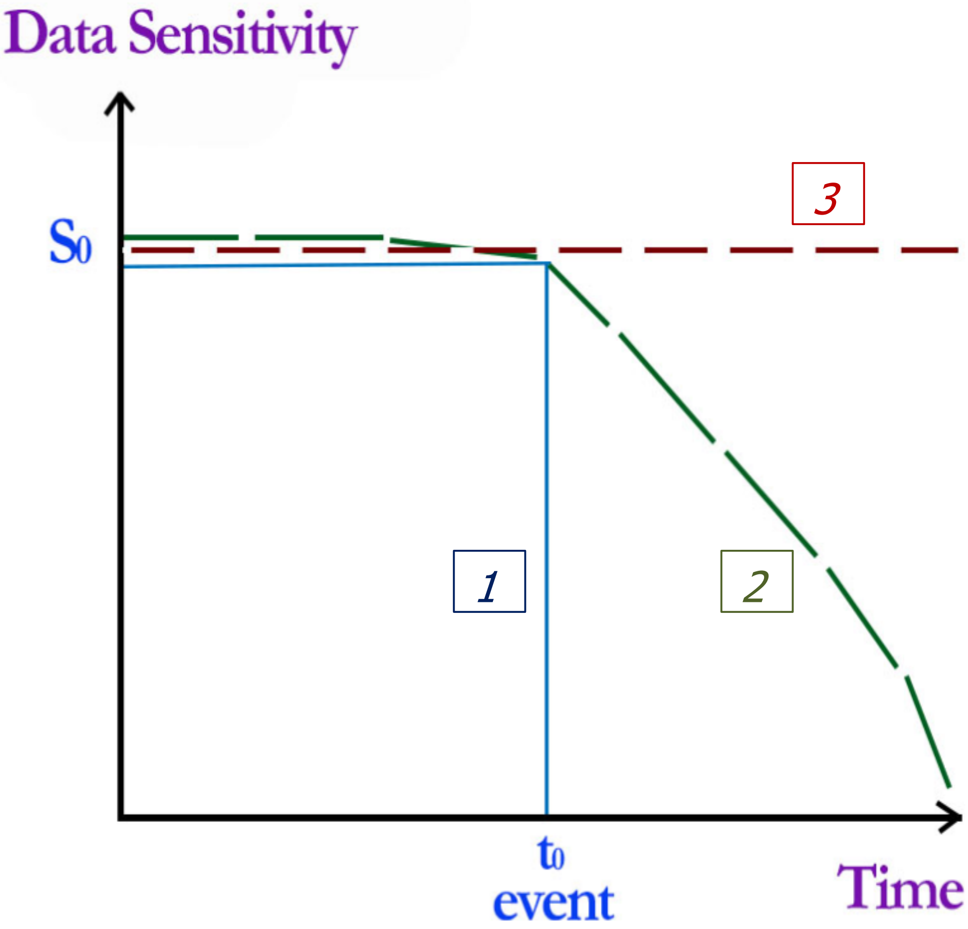


**Figure created by Ganapathy Mani, Purdue Univ.*

- 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)

Encrypted Database of ABs

- Collection agent gathers intelligence feeds (ABs)
- CryptDB is a proxy to a database server
  - Stores encrypted data (keywords, abstract of AB) and provides SQL query capability over encrypted data
  - Never releases decryption key to a database
  - When compromised, only ciphertext is revealed and data leakage is limited to data for currently logged in users
- Subscription API to provide methods for authorized access to data
- TF-IDF and Latent Semantic Indexing (LSI) retrieval models can be used for search engine
- Search query example:
  - ➢ "***insomnia diagnosis***" - find EHRs (ABs) of patients with diagnosis "insomnia" and retrieve data from them

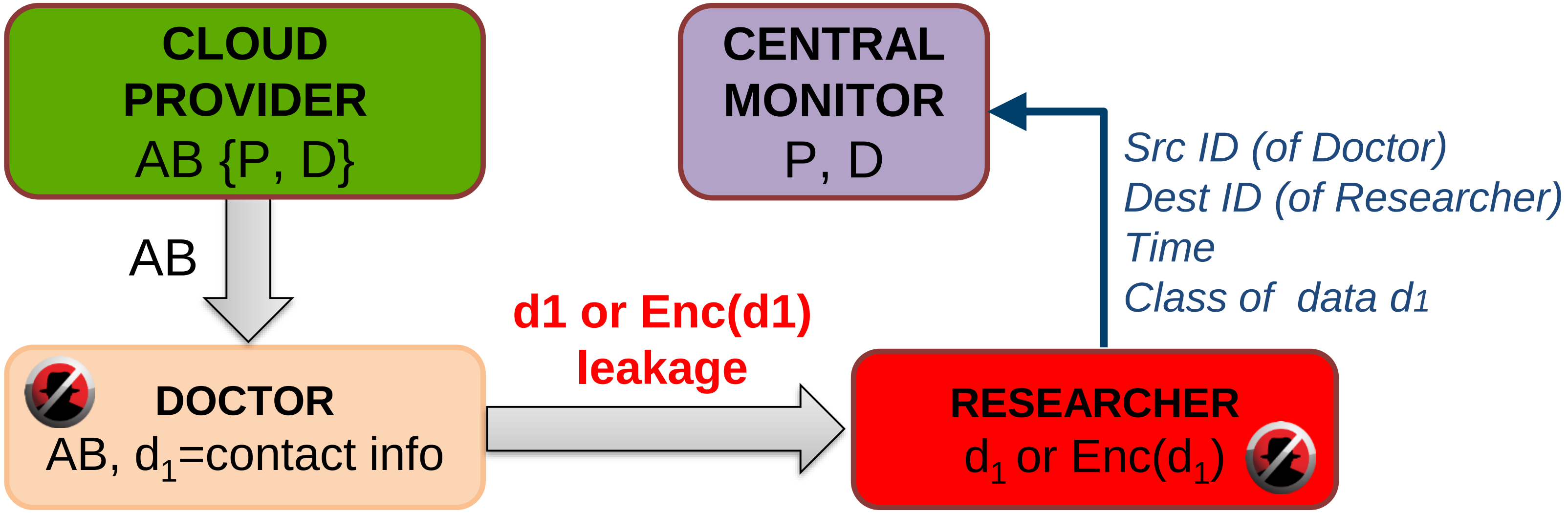
Result of weekly meetings with Dr. Leon Li, Jason Kobes, NGC

Experiments

- **Purpose:**
  - Identify details of what data, to whom, from where and when got leaked
  - Isolate data leakage
  - Identify vulnerabilities in the system
  - Assess data leakage damage
- **Input:**
  - Data (Electronic Health Record of a patient) stored in cloud in the form of Active Bundle
  - Access Control Policies
  - 3 services exchanging data: Doctor, Insurance, Researcher
- **Output parameters:**
  - Data Provenance Log
  - Sensitivity of Leaked Data
  - Vulnerability for data leakage
  - Timing of leakage
  - Extent of Damage

Experimental Setup

- **Experimental Setup:**
  - Three Node.JS services (Doctor, Insurance Agent, Researcher) exchanging data
  - Active Bundle containing data D (EHR of a patient) and access control policies P, hosted by cloud provider
  - Central Monitor with database of obligations (policies)
  - Doctor authorized to access contact info of a patient leaks it to Researcher who is not authorized to access it



CLOUD PROVIDER
AB {P, D}

CENTRAL MONITOR
P, D

AB

Src ID (of Doctor)
Dest ID (of Researcher)
Time
Class of data d1

d1 or Enc(d1) leakage

DOCTOR
AB, d_1=contact info

RESEARCHER
d_1 or Enc(d_1)

Deliverables

- **Prototype implementation:**
    - Data Leakage prototype
    - Active Bundle Module
        - AB implementation as an executable JAR file
        - AB API implementation using Apache Thrift RPC framework
        - Policy specification in JSON and evaluation using WSO2 Balana

Source code: http://github.com/Denis-Ulybysh/absoa16

- **Documentation:**
    - Deployment and user manual
    - Demo video *"Data dissemination/provenance in untrusted cloud"*

Ongoing Tasks

- Implementation of data leakage based on obligations and watermarks
- Deploy CryptDB storing encrypted AB-related data and providing SQL query capability over encrypted data
- Running AB in isolated container (e.g. Linux Docker)
- Experiments:
  - Tampering attacks on services (code injection etc.)
  - Man in the middle attacks (using an intercepted active bundle)
  - Attacks against data privacy (trying to bypass active bundle protection mechanism)
  - Tampering attacks on active bundle's policies and code
  - Cloud experiments: Framework scalability on industry standard cloud platforms (e.g. Amazon EC2)

- Integration with NGC projects:
  - WaxedPrune (with Donald Steiner/Jason Kobes/Leon Li of NGC)
  - CURATE 2 (with Jason Kobes)

Presentations and Publications

1. “Privacy-Preserving Data Dissemination and Data Leakage Detection in SOA” , D. Ulybyshev, B. Bhargava, L. Li, D. Steiner, J. Kobes, H. Halpin, M. Villarreal and R. Ranchal. *Submitted for ICDE-2017*
2. “Policy-based Distributed Data Dissemination,” R. Ranchal, D. Ulybyshev, P. Angin, and B. Bhargava. *CERIAS Security Symposium, April 2015* ***(Best poster award)***
3. “Authentication of User’s Device and Browser for Data Access in Untrusted Cloud,” D. Ulybyshev, B. Bhargava, L. Li, J. Kobes, D. Steiner, H. Halpin, B. An, M. Villarreal, R. Ranchal. *CERIAS Security Symposium, April 2016*.
4. "Cross-Domain Data Dissemination and Policy Enforcement", R. Ranchal, PhD Thesis, Purdue University, June 2015.
5. “End-to-End Security in Service-Oriented Architecture,” Mehdi Azarmi. *PhD Thesis*, Purdue University, April 2016.
6. “Consumer Oriented Privacy Preserving Access Control for Electronic Health Records in the Cloud,” R. Fernando, R. Ranchal. B. An, L. Ben Othmane, B. Bhargava. Submitted to *IEEE CLOUD 2016*.
7. “A Self-Cloning Agents-based Model for High Performance Mobile-Cloud Computing,” P. Angin, B. Bhargava, and Z. Jin. *IEEE CLOUD 2015*.

THE VALUE OF PERFORMANCE.

NORTHROP GRUMMAN

Secure/Resilient Systems

Moving Target Defense (MTD) Solution

CYBERSECURITY RESEARCH Consortium

Outline

- Collaboration with NGC IRADS
- Moving Target Defense (MTD)
- State of the Art and Limitations
- Research Approach
- Proposed Solution
- Benefits of the Proposed Solution
- Components of the Solution
- Research Tasks
- MTD Framework Prototype
- Experiments
- Future Directions

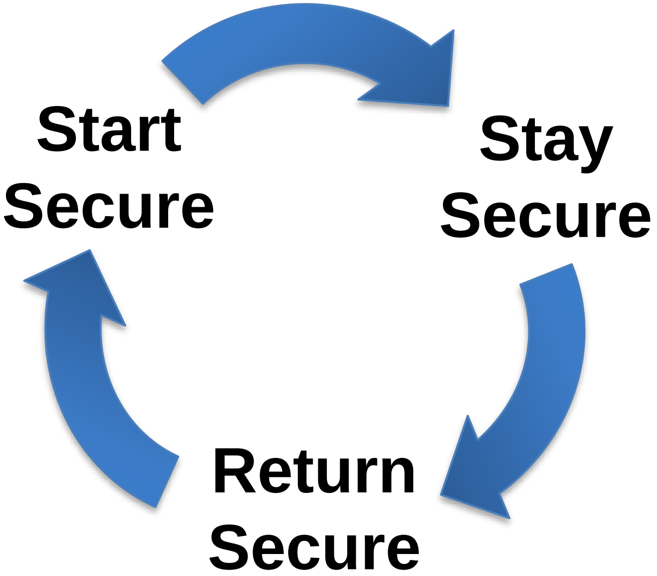
Collaboration with NGC IRADS

- During the TechExpo 2016: Discussion with Daniel Goodwin and Frank Wilson of NGC
- Contribute on the following IRADs:
  - Cyber Resilient Systems: Start Secure, Stay Secure and Return Secure (Daniel Goodwin)
  - Enterprise Resiliency (Frank Wilson)

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

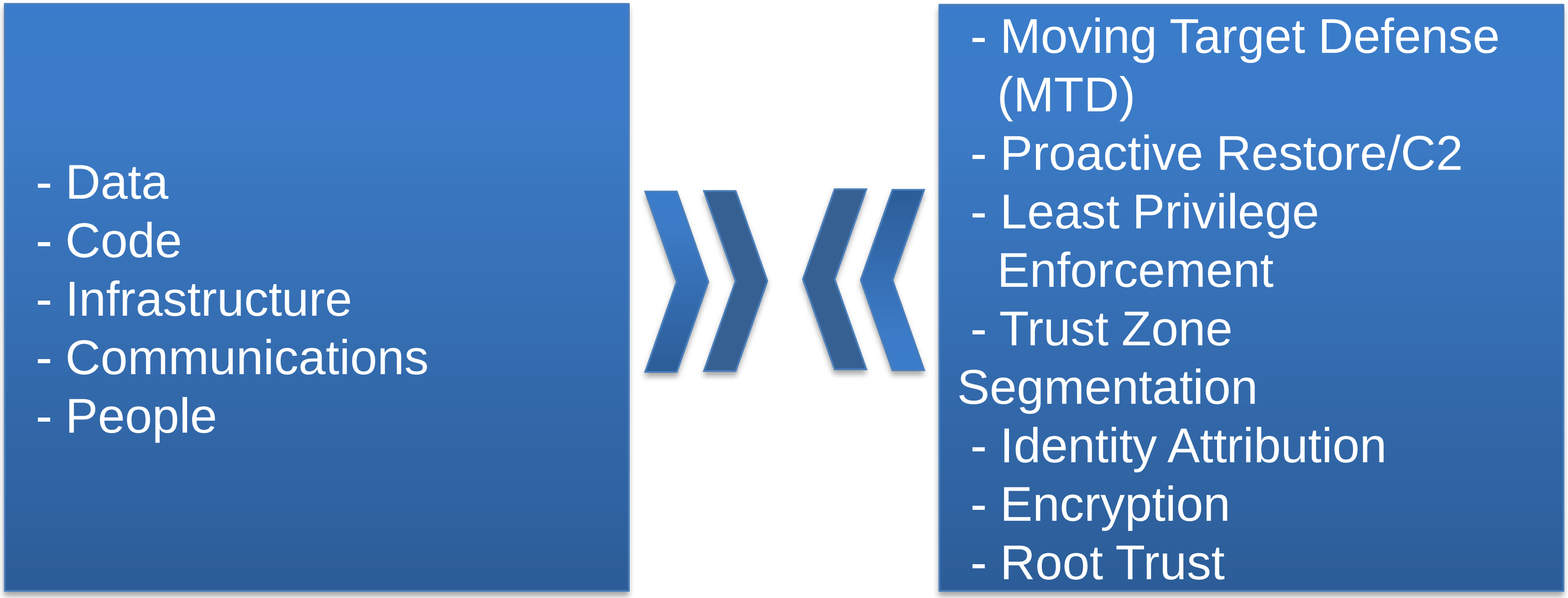
- The system starts with trusted components
- Continue to operate maintaining level of trust
- Return to trusted state case of an event



[1] NGC Cyber Resilient Systems IRAD (Daniel Goodwin)

Collaboration with NGC IRADS

- **Cyber Resilient Systems IRAD[1]:**



Attack Vectors

- Data
- Code
- Infrastructure
- Communications
- People

Resilient Approaches

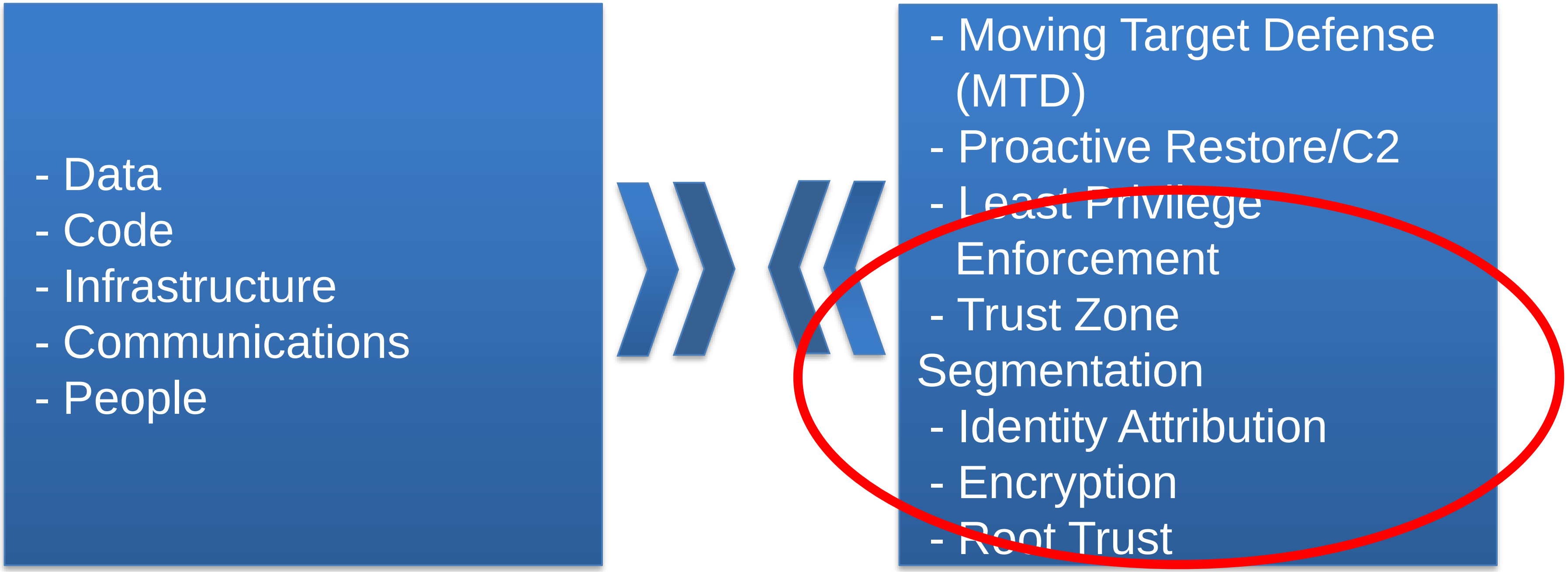
- Moving Target Defense (MTD)
- Proactive Restore/C2
- Least Privilege Enforcement
- Trust Zone Segmentation
- Identity Attribution
- Encryption
- Root Trust

[1] NGC Cyber Resilient Systems IRAD (Daniel Goodwin)

Collaboration with NGC IRADS

- **Cyber Resilient Systems IRAD[1]:**

Attack Vectors **Resilient Approaches**



- Data
- Code
- Infrastructure
- Communications
- People

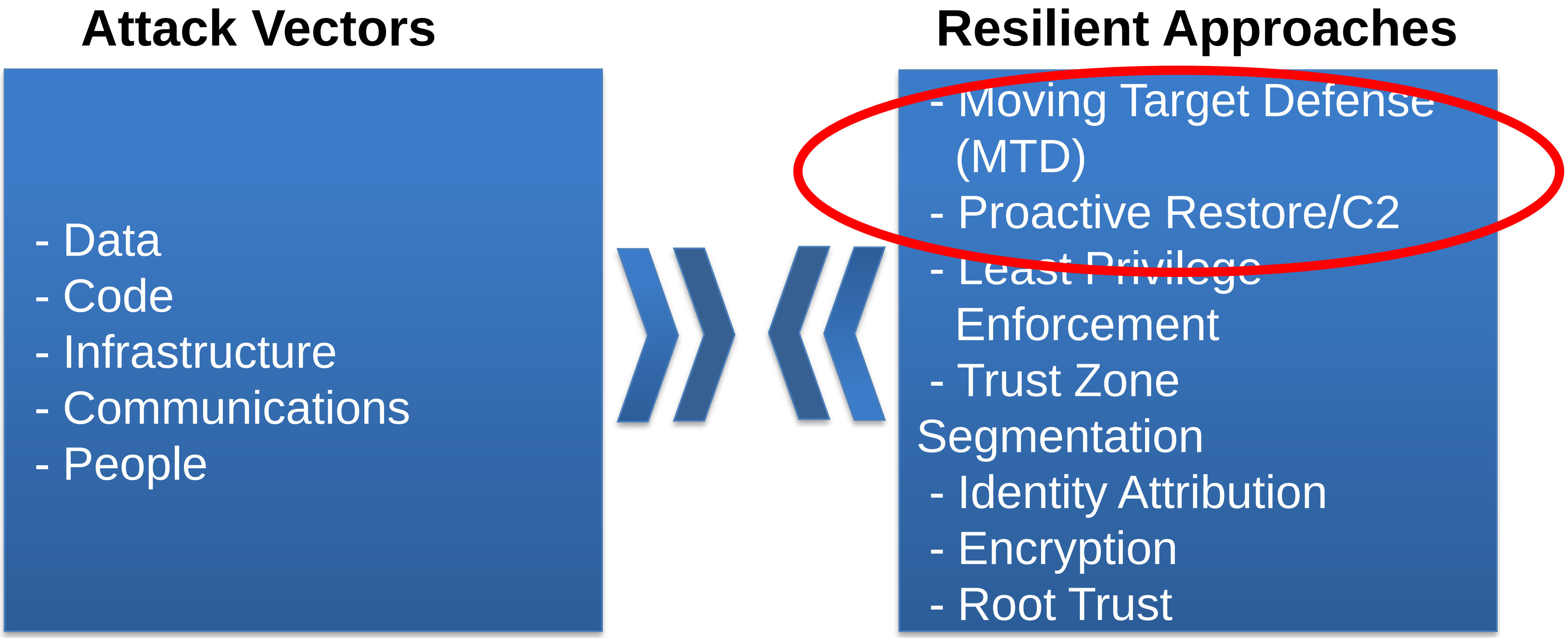
- Moving Target Defense (MTD)
- Proactive Restore/C2
- Least Privilege Enforcement
- Trust Zone Segmentation
- Identity Attribution
- Encryption
- Root Trust

Approaches followed in our **Active Bundle (AB)[8]** solution

[1] NGC Cyber Resilient Systems IRAD (Daniel Goodwin)

Collaboration with NGC IRADS

- **Cyber Resilient Systems IRAD[1]:**



Attack Vectors

- Data
- Code
- Infrastructure
- Communications
- People

Resilient Approaches

- Moving Target Defense (MTD)
- Proactive Restore/C2
- Least Privilege Enforcement
- Trust Zone Segmentation
- Identity Attribution
- Encryption
- Root Trust

Related to the proposed **Moving Target Defense (MTD)[4,5,6]** solution

[1] NGC Cyber Resilient Systems IRAD (Daniel Goodwin)

Collaboration with NGC IRADS

- **Enterprise Resiliency IRAD**[2]:
  - 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

[2] NGC Enterprise Resiliency IRAD (Frank Wilson)

Collaboration with NGC IRADS

- **Enterprise Resiliency IRAD[2]**:
  - 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

Directed related to the proposed **Moving Target Defense (MTD)[4,5,6]** solution

[2] NGC Enterprise Resiliency IRAD (Frank Wilson)

Collaboration with NGC IRADS

- The proposed **Moving Target Defense (MTD)** solution introduces resiliency and adaptability to the system through live monitoring, which transforms systems to be able to adapt and self-heal when ongoing attacks are detected

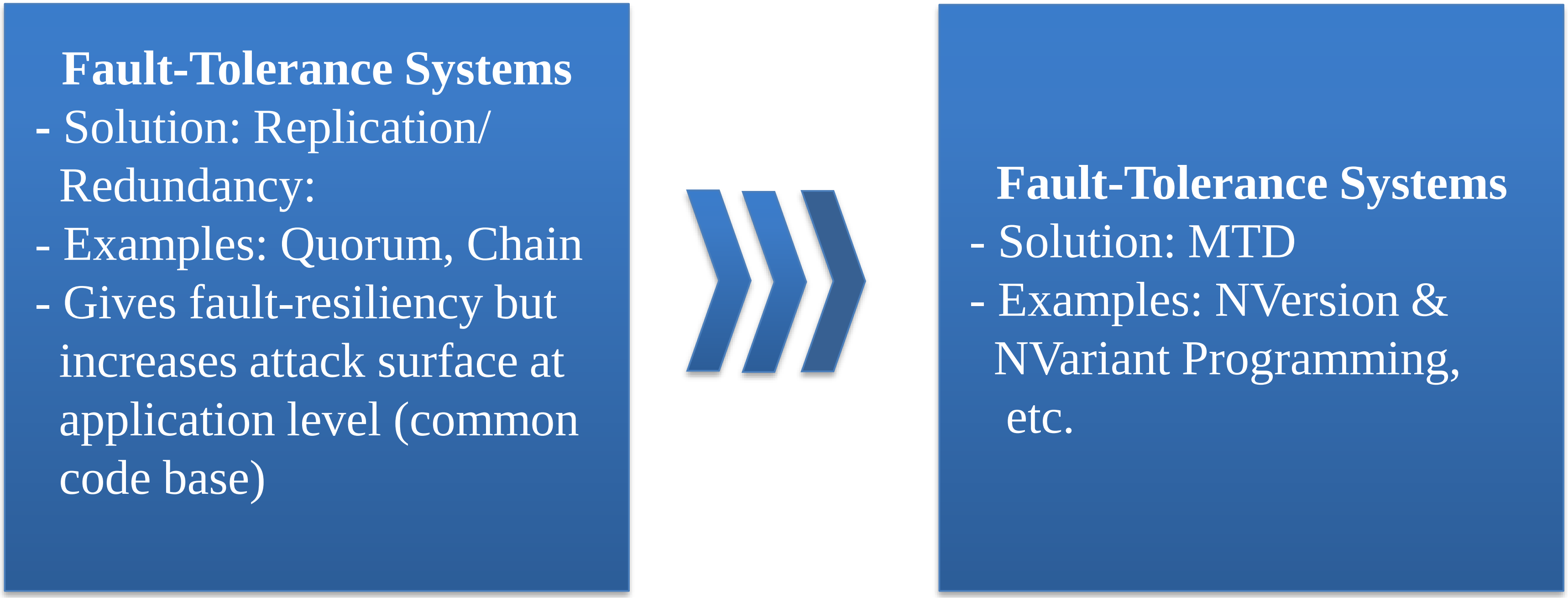
Moving Target Defense (MTD)

- **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

Threat Avoidance Techniques!

State of the Art and Limitations

MTD: Diversification and Randomization



Fault-Tolerance Systems

- Solution: Replication/ Redundancy:
- Examples: Quorum, Chain
- Gives fault-resiliency but increases attack surface at application level (common code base)

Fault-Tolerance Systems

- Solution: MTD
- Examples: NVersion & NVariant Programming, etc.

System level attacks

- Solution: MTD
- Examples: ASR, ISR, Inherent Issue- Low Entropy (JIT-ROP)
- New solutions: Isomeron, IP-Hopping in space

State of the Art and Limitations

- The traditional defensive security strategy for distributed systems is to prevent attackers from gaining control of the system using well established techniques: Replication/Redundancy, Encryption, etc.
- **Limitation**: Given sufficient time and resources, existing defensive methods can be defeated

State of the Art and Limitations

- The state of the art MTD solutions focus on randomization and diversification in particular layers of the system
- **Limitation**: Do not protect the entire host

Research Approach

- **“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, antifragility and adaptability of the system
  - Unified generic MTD framework that enables reasoning about behavior of deployed systems on cloud platforms

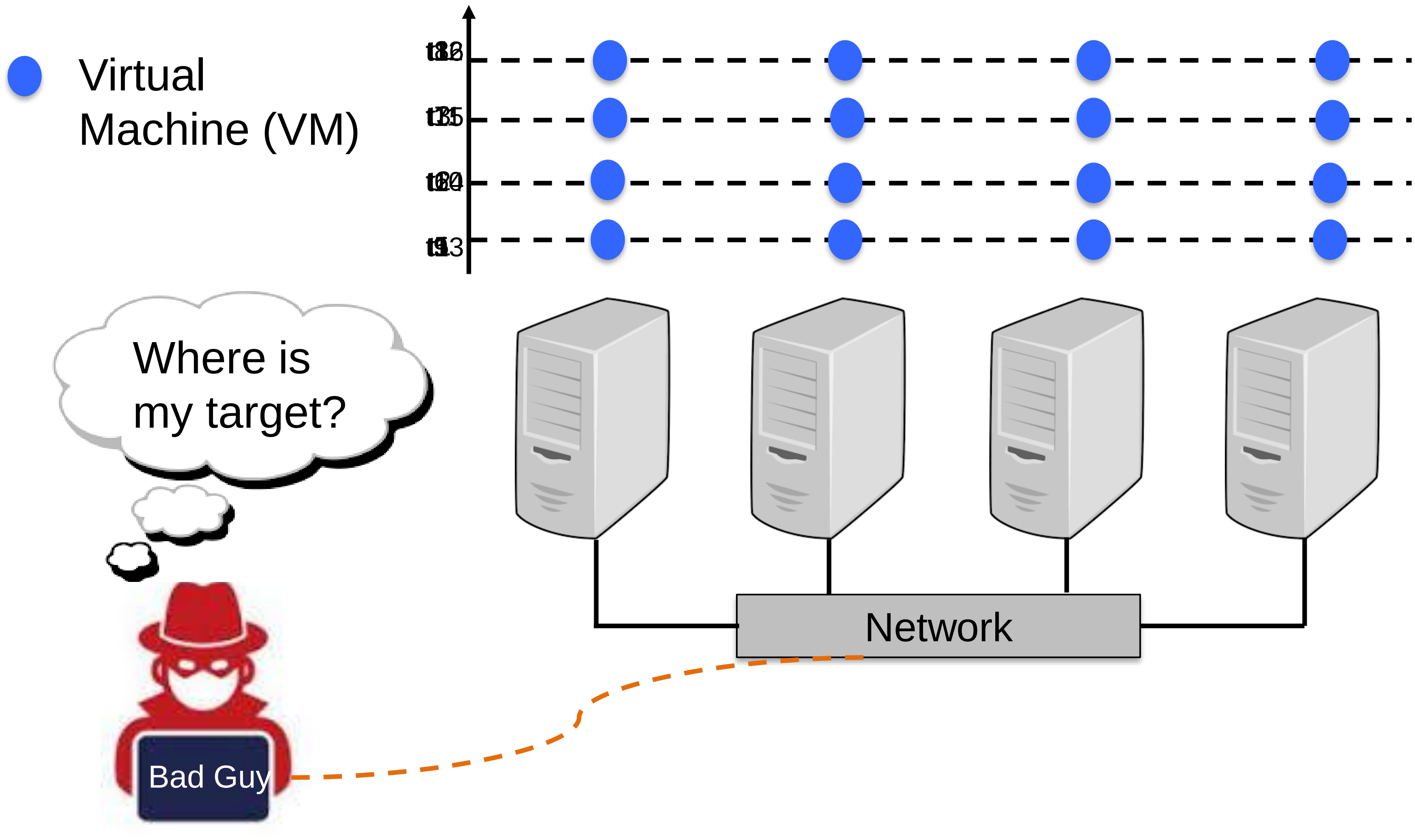
Proposed Solution

- Attack-resilient virtualization-based framework
- Aims to reduce the need to continuously fight against attacks by decreasing the gain-loss balance perception of attackers.
- Narrows the exposure window of a node to such attacks, which increases the cost of attacks to systems and lowers the likelihood of success and the perceived benefit of compromising it

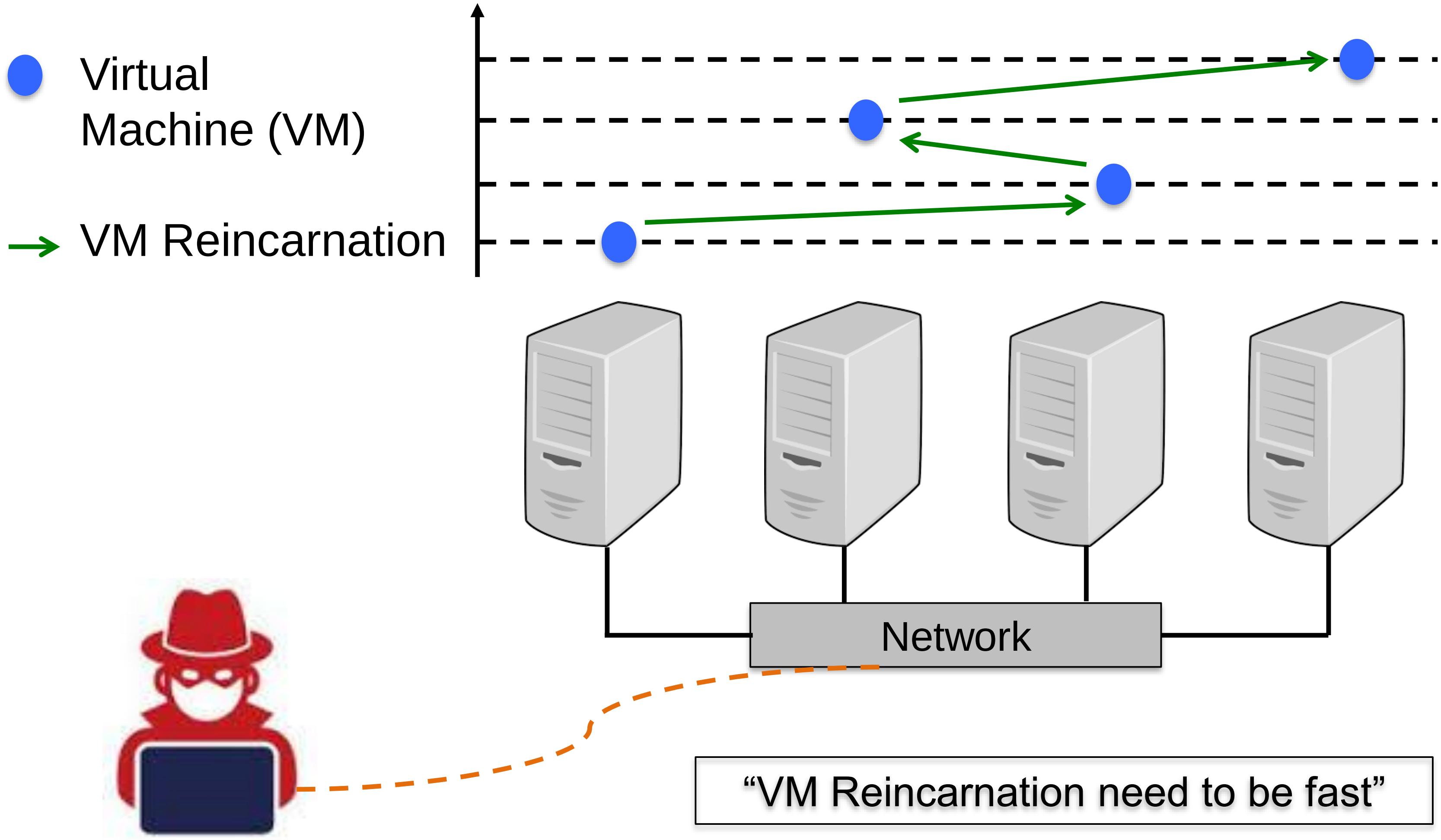
Proposed Solution

- The reduction in the vulnerability window of nodes is mainly achieved through three steps:
  - Partitioning the runtime execution of nodes in time intervals
  - Allowing nodes to run only with a predefined lifespan (as low as a minute) on heterogeneous platforms (i.e. different OSs)
  - Proactively monitoring their runtime below the OS

Proposed Solution



Proposed Solution



● Virtual Machine (VM)

→ VM Reincarnation

“VM Reincarnation need to be fast”

Proposed Solution

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

- Nodes run on a given computing platform only for a controlled period of time
- The running time is chosen in a way that successful ongoing attacks become ineffective

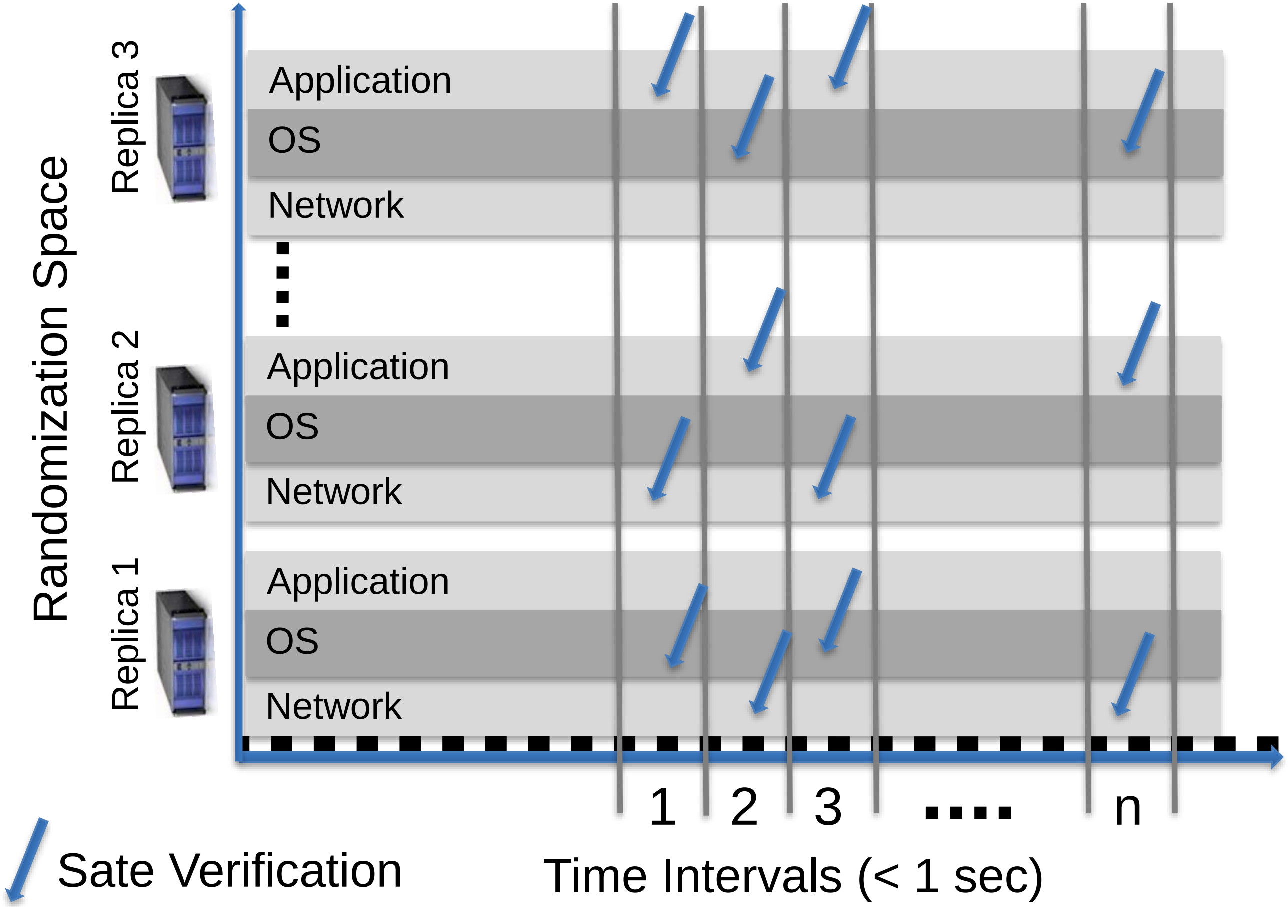
Proposed Solution

NORTHROP GRUMMAN

- The new fresh machine will integrate to the system and continue running the application after getting up to date
- Two research directions: **Stateless** vs. **Stateful** applications
- Stateless application examples: BFT and pub/sub systems

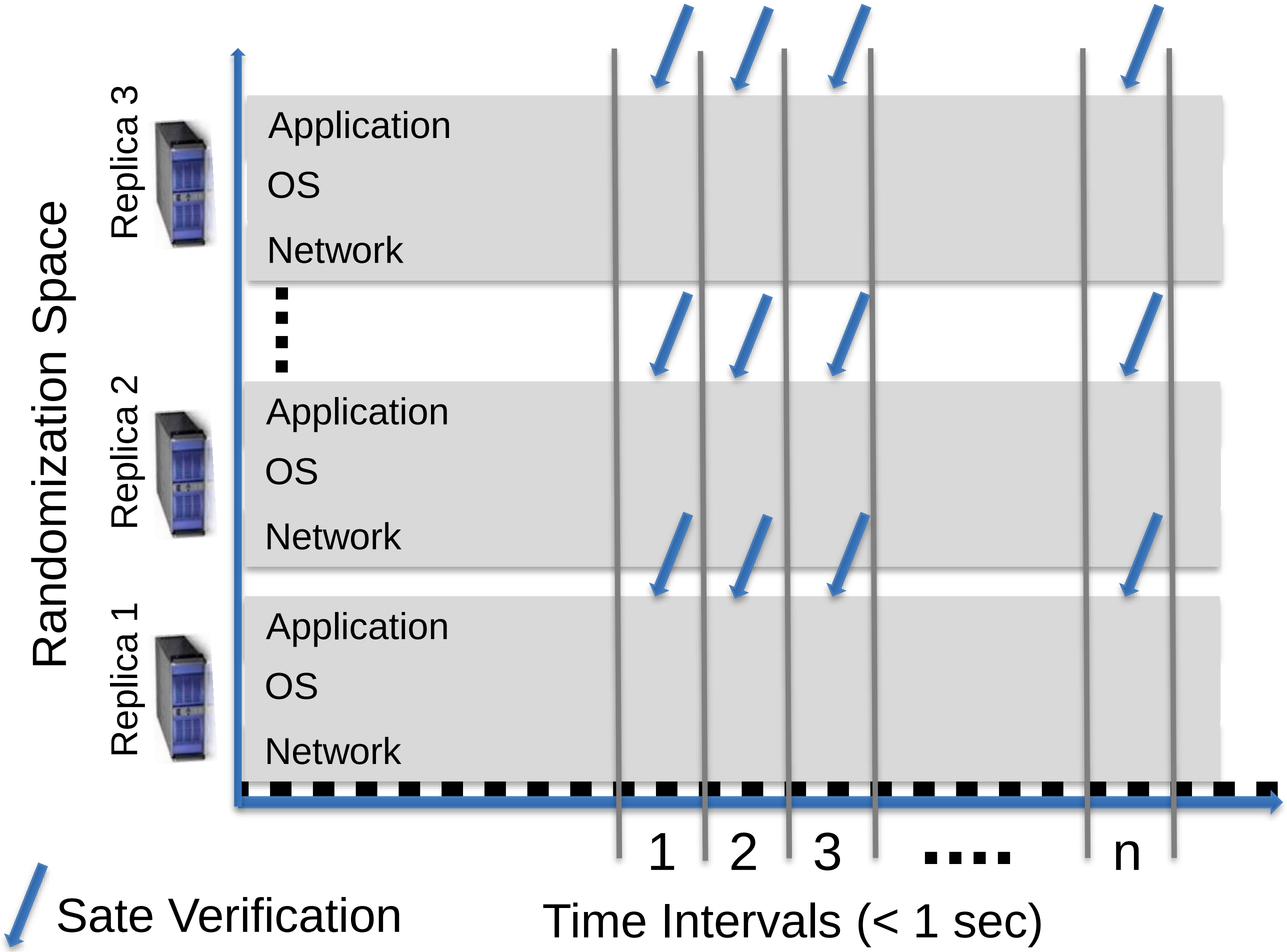
Benefits of the Proposed Solution

- **State of the Art System View:**



Benefits of the Proposed Solution

- **Proposed Solution System View:**



Components of the Framework

- The framework will consist of the following four components:
    - Proactive Virtual Reincarnation (ViRA)
    - Proactive Monitoring
    - SDN Network Dynamics
    - Systems States and Application Runtime
- The framework will protect the whole stack; not only particular layers

Components of the Framework

- **Proactive Virtual Reincarnation (ViRA):**
  - OpenStack: Widely adopted industry open source framework (key component: Nova)
  - Nova: The cloud management software of OpenStack, which provides virtual machines on demand
  - One approach is setting a fixed period of time for each machine and reincarnating them after that lifespan
  - Machines to be reincarnated are selected either in Round Robin or random fashion
  - Attacks can occur within the lifespan of each machine, which makes **proactive monitoring** mechanisms a crucial element component

Components of the Framework

- **Proactive Monitoring:**
  - Virtual Machine Introspection (VMI)
  - Open Source Prototype
  - Will trigger reincarnations when attacks are detected
  - Deployment Challenges: Semantic Gap, Interoperability Issues with Cloud Frameworks
  - **Solution Approach**[3]: Ahmed, N., and Bhargava, B. Towards Targeted Intrusion Detection Deployments in Cloud Computing. In the Int. Journal of Next-Generation Computing Vol. 6, No 2, IJNGC - JULY 2015.

Components of the Framework

- **SDN Network Dynamics:**
  - Based on Neutron, a component of Open Stack
  - Neutron: An Open Stack implementation of Software Defined Networks (SDN), which provides networking as a service
  - Fresh VM must keep the same network configuration as the vanishing VM

Components of the Framework

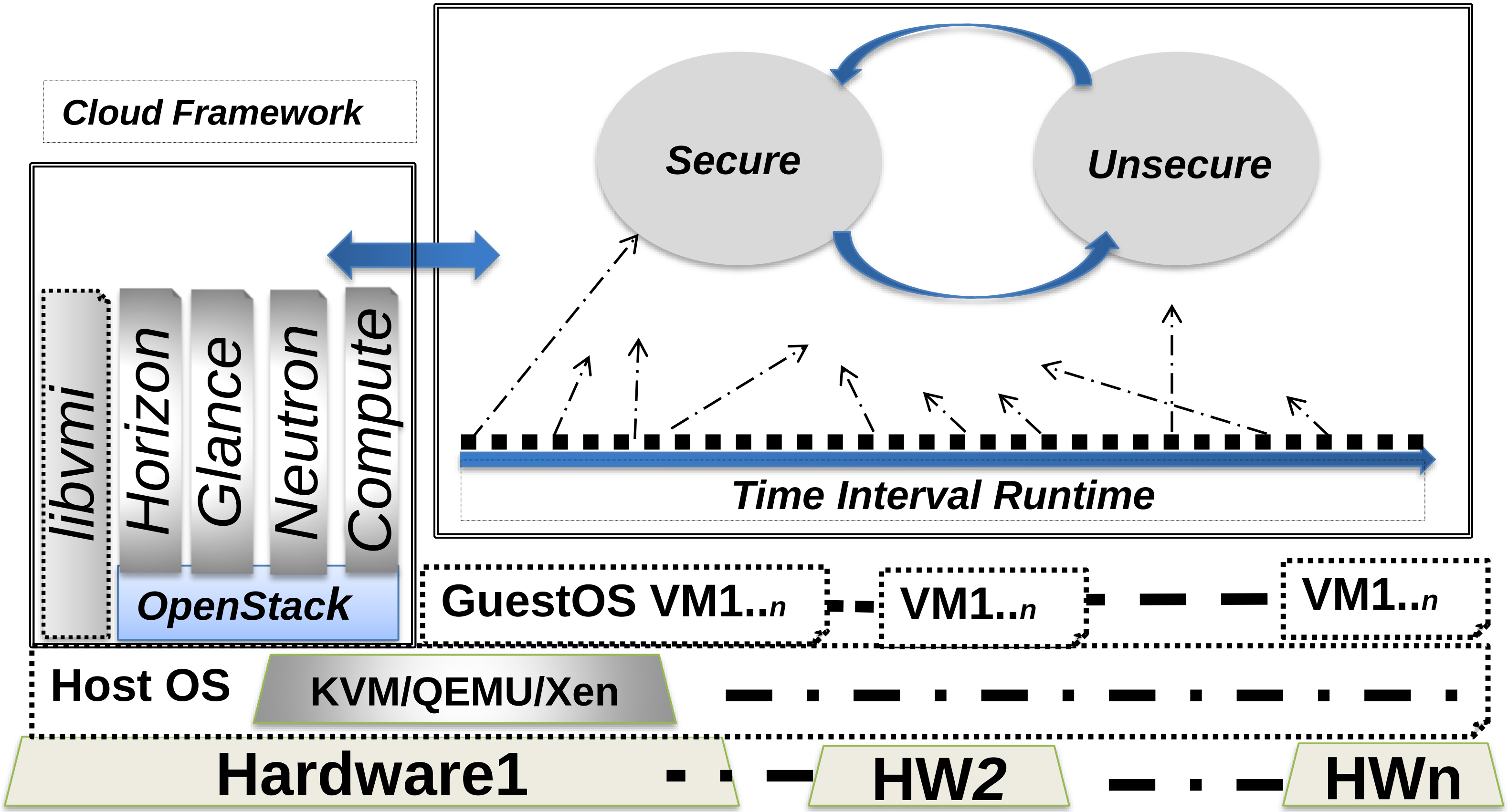
- **System States and Application Runtime:**
  - System State: The state of the system at any given time (Secure vs. Unsecure)
  - Application Runtime: At the application layer, we refresh and map one or more Apps/VMs (App1. . . Appn) to different platforms (Hardware1. . . HWn) in pre-specified time intervals, referred as time-interval runtime

Research Tasks

- Defining metrics to quantify effectiveness of system components (services, data, networks etc.)
- Defining costs of software-based reconfiguration/monitoring/healing of system components
- Developing models and mechanisms for optimized automated monitoring and reconfiguration of system architectures to achieve maximum possible resiliency with minimum operational cost
- Build a prototype of the MTD framework to conduct experiments to measure these metrics

Framework Prototype

MTD Framework



Framework Prototype

- The framework will be implemented using two main components: Nova, the cloud management software of OpenStack and the Software Defined Networking (SDN) tool Neutron
- Nova provides and de-provides virtual machines for cloud users on demand while Neutron provides networking as a service and runs on top of OpenStack
- The library *libvmi* will be installed for virtual machine introspection. Memory snapshots will be taken in time intervals at the host OS as applications to be protected run in the VM. The live memory data structure will be analyzed in real time and alarm any runtime integrity violation of the applications runtime as an intrusion

Experiments

- **Purpose:**
  - With the increased sophistication of attacks, is it possible to construct a generic attack-resilient framework for distributed systems (**stateless application**) with a combination of mobility and direction capabilities?
  - Taking into account the main components of the state of a virtual machine (i.e. memory and network), is it possible to build a generic resilient and application-agnostic platform without service (**stateful applications**) interruptions in the reincarnation process?

Experiments

- **Purpose:**
  - With the increased sophistication of attacks, is it possible to construct a generic attack-resilient framework for distributed systems (**stateless application**) with a combination of mobility and direction capabilities?
  - Taking into account the main components of the state of a virtual machine (i.e. memory and network), is it possible to build a generic resilient and application-agnostic platform without service (**stateful applications**) interruptions in the reincarnation process?

Experiments

- **Problem Statement:**
  - Verify the effectiveness of the virtual reincarnation process of the Moving Target Defense solution against system attacks, specifically attacks executed from compromised guest OS
  - Measure the impact of system attacks (compromised guest OS) on the Resilient Systems using the MTD solution

Experiments

- **Inputs (types of systems to test ViRA):**
  - Experiments will be run on BFT-SMaRt (synchronous system)
  - Experiments will be run on a basic Publish and Subscribe system (asynchronous system)
- **Inputs (attacks to test proactive monitoring):**
  - Attack 1: We mimic a node compromise by logging into the VM, stopping the victim process, and starting a malicious one with the same name but different functionality; presumably stealing data
  - Attack 2: We hijack the application by loading/injecting a shared library and diverting to make additional function calls without stopping the process

Experiments

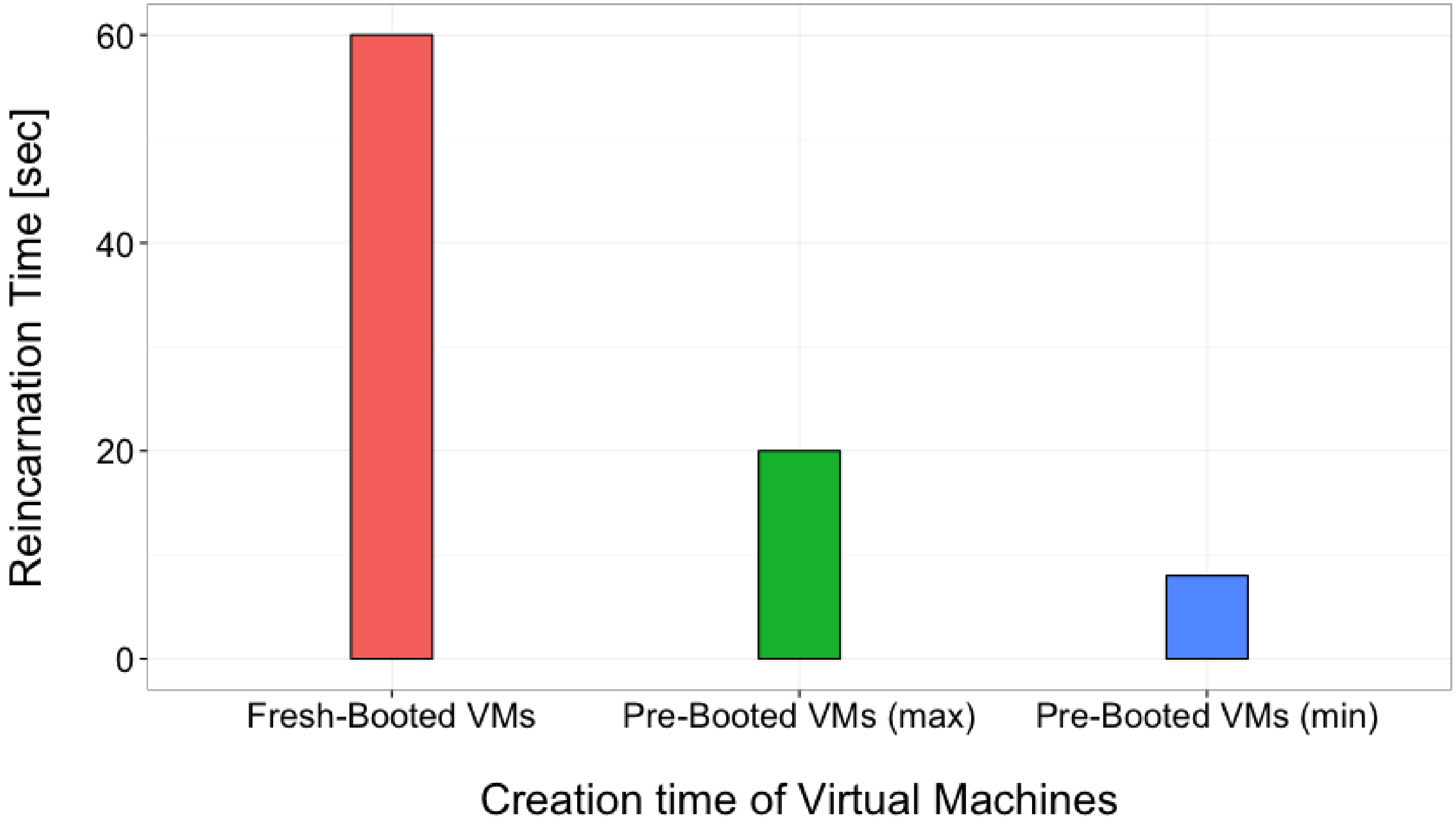
- **Outputs:**
  - Time required for virtual machine reincarnations triggered by ended lifespan for both BFT-SMaRt and Publish and subscribe systems
  - Time required for virtual machine reincarnations for both BFT-SMaRt and Publish and subscribe systems under system attacks
  - Detection rate of the Virtual Machine Instrospection technique.

Experiments

- **Methods:**
  - The framework will be implemented according to the prototype
  - Nodes will be reincarnated without attacks. Reincarnation times will be measured
  - Attacks will be launched. The library *libvmi* will be installed for virtual machine introspection. Effectiveness rate of the detection mechanism will be measured along with the time of reincarnation times under attack.

Experiments

- **Preliminary Results:**



Experiments

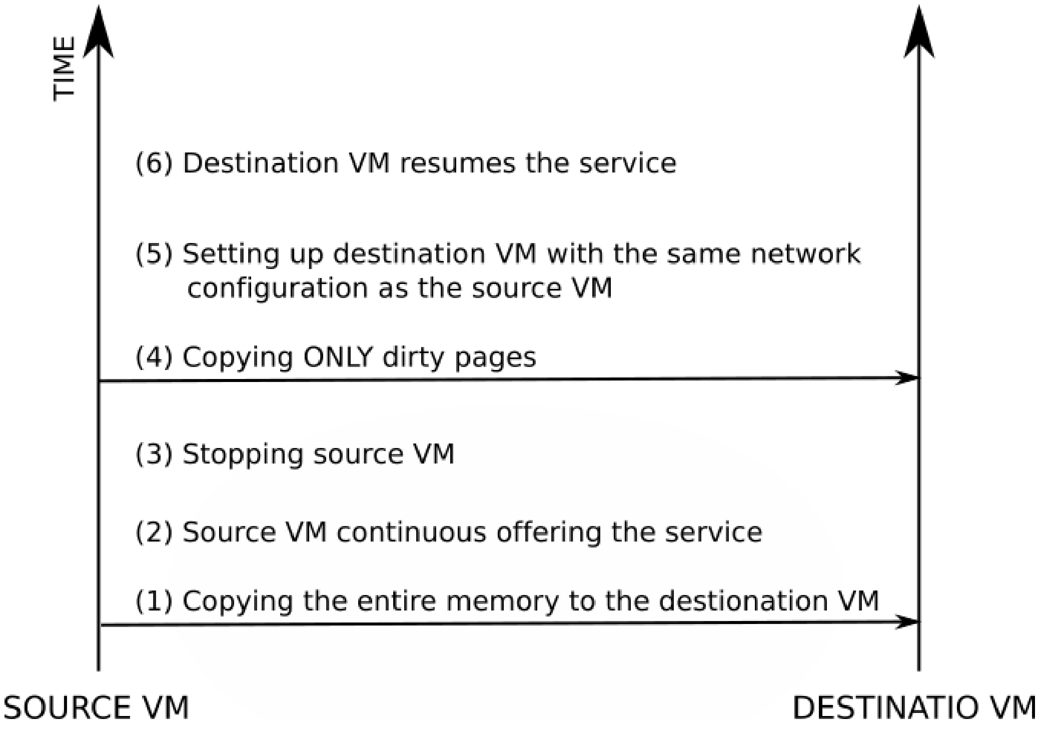
- **Observations:**
    - Impact on Cyber Resilient Systems and providing Enterprise Resiliency
    - Prototype will be extended to produce a demo for Tech Expo 2017
    - The work will closely be conducted with NGC to find feasible solutions for Cyber Resilient Systems and Enterprise Resiliency
    - It will include publishing the collaborative work and exploring funding opportunities at BAA in AFRL and NSF to secure funding and realize MTD as a viable solution

Future Directions

- **Test other stateless applications on the MTD framework:**
  - E.g.: Upright (Public and Subscribe System)
- **Stateful application support:**
  - Can we preserve the state of the virtual machine during the reincarnation process to make the solution application-agnostic?
  - Test the framework wit Secure SOA Services (stateful applications)

Future Directions

- **Stateful Application Support (Cont…):**
  - Virtual reincarnation must be fast. The source VM can continue running until the destination VM is ready to take over


TIME
(6) Destination VM resumes the service
(5) Setting up destination VM with the same network configuration as the source VM
(4) Copying ONLY dirty pages
(3) Stopping source VM
(2) Source VM continuous offering the service
(1) Copying the entire memory to the destionation VM
SOURCE VM
DESTINATIO VM

Presentations and Publications

1. NGC Cyber Resilient Systems IRAD (http://www.northropgrumman.com)
2. Enterprise Resiliency IRAD (http://www.northropgrumman.com)
3. Ahmed, N., and Bhargava, B. Towards Targeted Intrusion Detection Deployments in Cloud Computing. In the Int. Journal of Next-Generation Computing Vol. 6, No 2, IJNGC - JULY 2015.
4. N. Ahmed. Design, Implementation, and Experiments for Moving Target Defense. PhD Thesis, Purdue University, 2016.
5. N. Ahmed and B. Bhargava. From Byzantine Fault-Tolerance to Fault-Avoidance: An Architectural Transformation to Attack and Failure Resilience. To Appear in IEEE Transactions on Cloud Computing, TCC 2016.
6. N. Ahmed and B. Bhargava. Disruption-Resilient Publish/Subscribe: A Moving Target Defense Approach. The 6th International Conference on Cloud Computing and Services Science, CLOSER 2016.
7. N. Ahmed and B. Bhargava. Mayflies: A Moving Target Defense Framework for Distributed Systems. 3rd ACM workshop on MTD in conjunction with ACM Conference on Computer and Communications Security (CCS), Vienna, 2016.
8. R. Ranchal, D. Ulybyshev, P. Angin, and B. Bhargava. Policy-based Distributed Data Dissemination. *CERIAS Security Symposium, April 2015* ***(Best poster award)***

THE VALUE OF PERFORMANCE.

NORTHROP GRUMMAN

Thank you!

CYBERSECURITY
RESEARCH
Consortium