Uptane

Uptane++ Implementation on Robotic and Autonomous Systems

SOUTHWEST RESEARCH INSTITUTE®

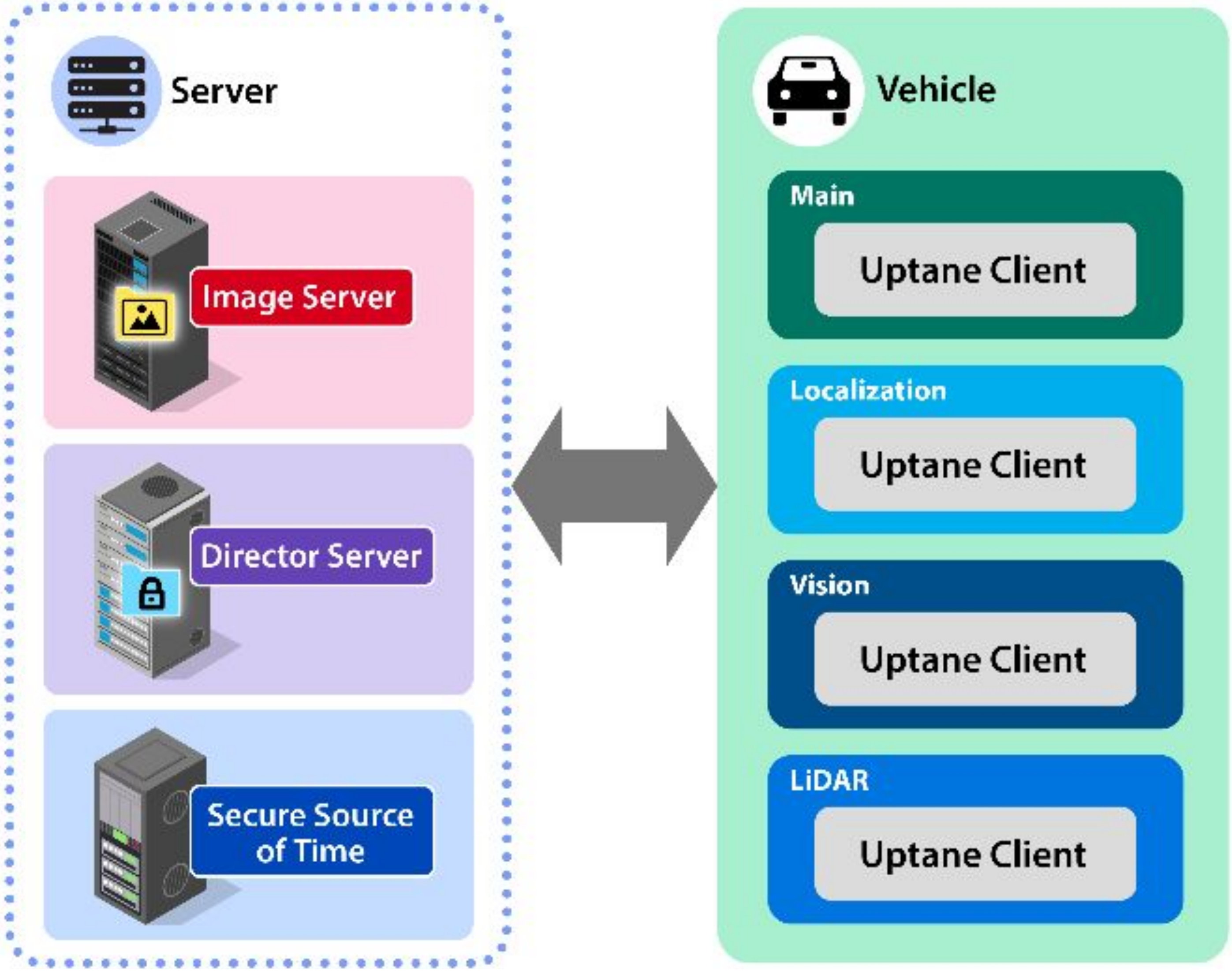
Cameron Mott

Uptane Alliance 2023

Robotic and Autonomous Systems

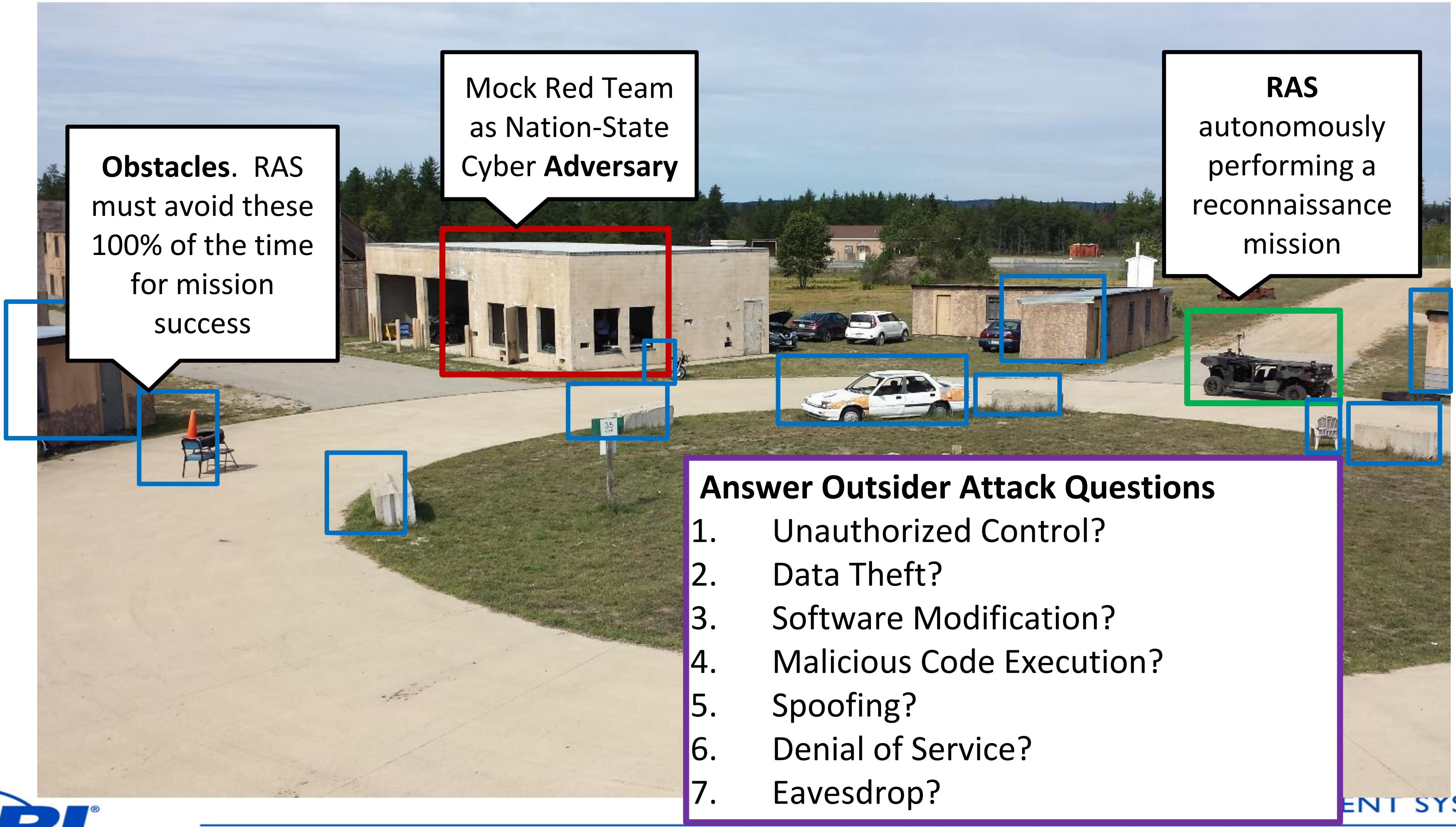
- RAS provide essential and life-saving capabilities to DoD forces
- Fully automated MRZR platform
  - Forward reconnaissance
  - Mission resupply
  - Sensor payload
- Advanced computing systems – architecture similar to EVs
- Tele-operation
- Waypoint following
- Off-road capable

System Architecture



CRASH

Operational Demo Concept: Outsider Threat



Obstacles. RAS must avoid these 100% of the time for mission success

Mock Red Team as Nation-State Cyber **Adversary**

RAS autonomously performing a reconnaissance mission

Answer Outsider Attack Questions

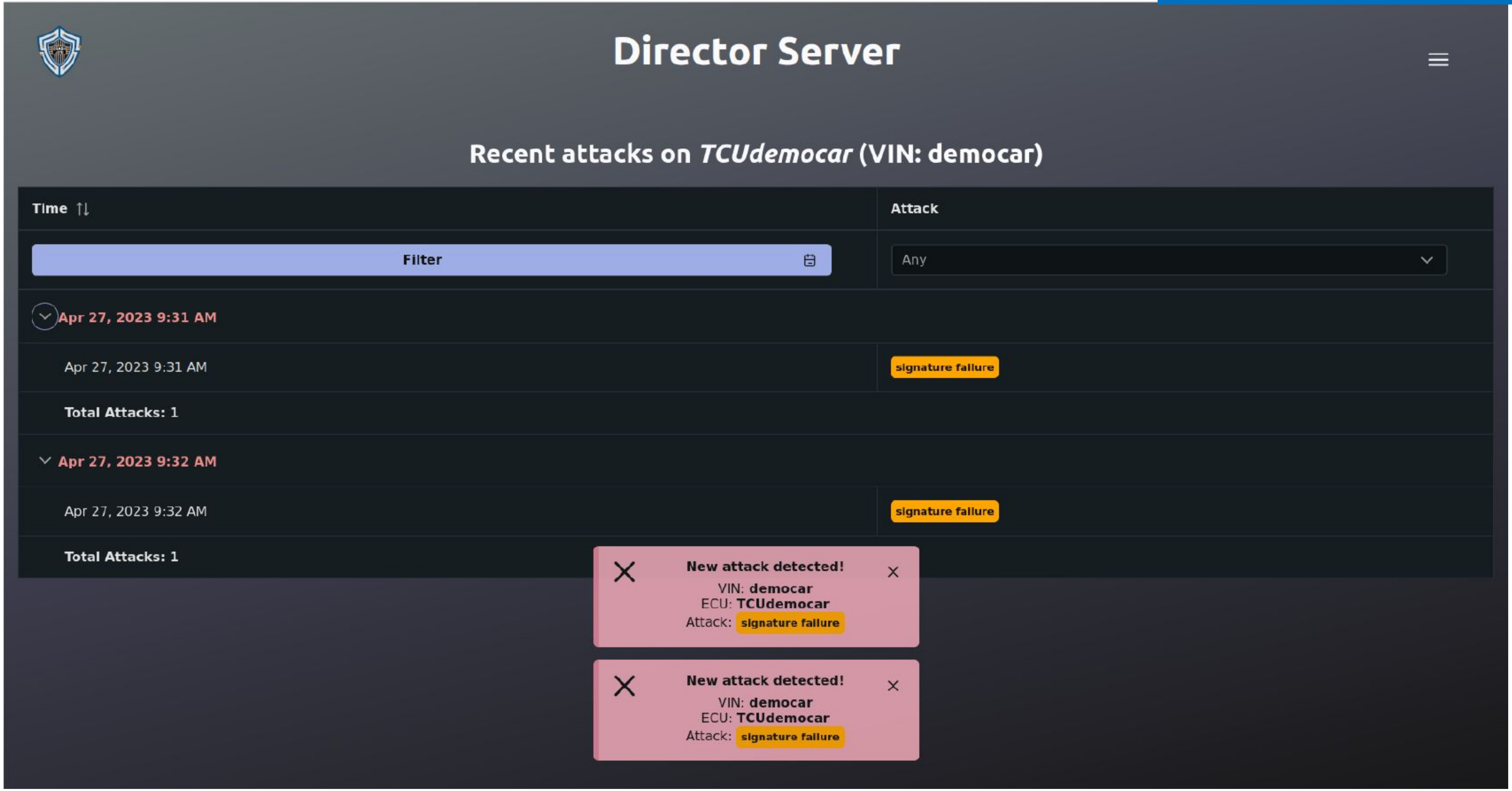
1. Unauthorized Control?
2. Data Theft?
3. Software Modification?
4. Malicious Code Execution?
5. Spoofing?
6. Denial of Service?
7. Eavesdrop?

Attacks and Protections

Attack	Goal Description	Protections
Eavesdropping	Extract information from updates sent from the repository to an ECU	Encrypted communications and files
Denial of Service	Prevent server and client from communicating effectively	Timeliness of file transfers Vehicle installation reports
Partial Installation	Prevent parts of the update from reaching the client	“All-or-nothing” updates Vehicle installation reports
Infinite Data	Send a large enough amount of data to an ECU to exhaust ECU storage	Download length limits
Incorrect Installation	Have a legitimate ECU or vehicle access an update that is not intended for it	Use of vehicle and hardware identifiers Files encrypted per vehicle
Rollback	Send out a previously deployed update to an ECU when a newer update exists	Monotonically increasing version numbers
Mixed Bundle	Force an ECU to install incompatible versions of software update files	Monotonically increasing version numbers Signature validation
Arbitrary Edit	Edit updates sent from the repository to an ECU and make it non-functional	Multiple signature and hash validation
Injection	Add their own functionality to an update through injecting data into the update package	Multiple signature and hash validation Key quorum with offline keys
Insider Attack	Use insider keys/credentials to send malicious update package	Multiple signature validation Key quorum with offline keys

Attack alerts

Uptane



Director Server

Recent attacks on *TCUdemocar* (VIN: democar)

Time ↑↓	Attack
Filter	Any
Apr 27, 2023 9:31 AM	
Apr 27, 2023 9:31 AM	signature failure
Total Attacks: 1	
Apr 27, 2023 9:32 AM	
Apr 27, 2023 9:32 AM	signature failure
Total Attacks: 1	

New attack detected!
VIN: **democar**
ECU: **TCUdemocar**
Attack: signature failure

New attack detected!
VIN: **democar**
ECU: **TCUdemocar**
Attack: signature failure

Secure Software Updates

Uptane

- Software on RAS need to be updated
  - Functionality improvements
  - Bug fixes
  - Cybersecurity updates
  - Mission-specific capabilities
- Update mechanisms can be leveraged as a threat vector
  - Operating in hostile environments
  - Existing adversaries
- Persistent attacker
- Protect and secure updates
  - in-transit
  - at-rest

Accomplishments

Uptane

- Implemented Uptane clients and server back-end in C++
- Server
  - Middle-ware written in Rust
  - Front-end in React
- Dramatically improved security
  - Encrypted data-in-transit
  - Microkernel segmentation and containerization
  - Secure software updates
  - Data policy engine and enforcement
- Demonstration in a mock field environment
- Risk management framework artifacts
- Training and knowledge transfer hand-off
  - DevSecOps
  - External deployer trained on the system within 1 day and successfully leveraged the capabilities of the secure update solution

Complexities

- Multiple virtual machines
- Hierarchical filesystems
  - File management – addition, removal, untracked files
- Support for dynamic files (logs, configurations)
- Integration with the seL4 microkernel
- Key management

Improvements to Standard

- Adjustments to standard based on implementation efforts
- Incorporate mapping between requirements and attack preventions
- Provide metrics of implementation resource sizes

Adjustments to standard

Uptane

Based on Uptane 2.0.0

1. Clarity regarding VIN in metadata and verification
    a. 5.2.3.1. Metadata about images "If there are no images included in the Targets metadata from the Director repository, then the metadata SHALL include a vehicle identifier in order to avoid a replay attack."
    b. Nowhere in verification steps of targets metadata does it mention to check for the vehicle identifier
2. Clarify if image verification hash / length checking includes encrypted versions of the images
    a. This is clearly stated how to be done in step 10.2.1 of 5.4.4.2. Full verification, but not so clear in step 7 of 5.4.3.4. Verify image.
3. Clarify checking release counters
    a. What is the "previous Targets metadata" that is used to check the release counters?
        i. From the context, it seems like the Standard is implying that the Director Targets is the current, Image Targets for the previous
    b. What is the difference in the checks in matching Targets verification and image verification?
        i. How is this a check for matching Targets?
4. Clarify under what conditions we would have no Targets metadata about an image, while still having targets metadata for an ECU identifier
    a. Step 4 of 5.4.3.3. Download latest image
    b. We don't have a way to sensibly perform this check
5. More explicit guidance as to when to replace the previous metadata file
    a. This is done clearly in section 5.4.4.3. How to check Root metadata (specifically step 2.5)
    b. We have a system where we discard new metadata if update is aborted, but I could not find anything that fully covers this
    c. Step 4 of 5.4.3.3. Download latest image says to retain metadata on some condition, others say to discard on some condition but it is not the most clear
6. Clearer distinctions between Image server and Director server APIs
    a. In deployment considerations, the Image server does not specify to have a private API (possibly because the standard says "The Image repository SHALL provide a method for authorized users to upload images and their associated metadata", but this is unclear
    b. Deployment considerations says Director private API should allow OEM to upload images

Mapping

Attack	Goal Description	Protections
Eavesdropping	Extract information from updates sent from the repository to an ECU	Encrypted communications and files
Denial of Service	Prevent server and client from communicating effectively	Timeliness of file transfers Vehicle installation reports
Partial Installation	Prevent parts of the update from reaching the client	“All-or-nothing” updates Vehicle installation reports
Infinite Data	Send a large enough amount of data to an ECU to exhaust ECU storage	Download length limits
Incorrect Installation	Have a legitimate ECU or vehicle access an update that is not intended for it	Use of vehicle and hardware identifiers Files encrypted per vehicle
Rollback	Send out a previously deployed update to an ECU when a newer update exists	Monotonically increasing version numbers
Mixed Bundle	Force an ECU to install incompatible versions of software update files	Monotonically increasing version numbers Signature validation
Arbitrary Edit	Edit updates sent from the repository to an ECU and make it non-functional	Multiple signature and hash validation
Injection	Add their own functionality to an update through injecting data into the update package	Multiple signature and hash validation Key quorum with offline keys
Insider Attack	Use insider keys/credentials to send malicious update package	Multiple signature validation Key quorum with offline keys

Uptane

Uptane Description of Threats	Attacks	Our Description	Attacker Capabilities	Minimum Key Compromises	Notes	Difficulty Rank
Read Updates	Eavesdrop Attack	attacker can extract information from updates sent from the repository to an ECU	MITM	ECU image decryption key if encrypted		8/4
Deny Install of Update	Denial of Service Attack	an attacker can block network traffic and prevent an ECU from receiving updates	MITM (block traffic outside vehicle)	-	Cannot be prevented if MITM but can be detected (lack of response from primary, version manifest)	7
	Partial Installation Attack	an attacker can allow only part of an update to download by dropping selected traffic	MITM (block traffic inside vehicle)	ECU Private Key – and sign an incorrect manifest	Cannot be prevented if MITM but can be detected with the vehicle version manifest listing installed images	6?
	Arbitrary Edit Attack	an attacker can edit updates sent from the repository to an ECU	MITM (edit packets)	-	**assuming goal is to edit for update rejection	6?
Interfere with ECU Function	Infinite Data Attack	attacker can send a large enough amount of data to an ECU that the ECU will run out of storage.	MITM (intercept packets, edit packets, send packets) Cryptographic/signing ability Possibly remote exploit	Director Target Key (1) and Majority of Image Targets keys (ex. 3/4)	Easiest to succeed would be to modify both targets metadata stated size of image and sign Image targets are offline keys	5/?
	Incorrect Installation Attack	a legitimate vehicle (or ECU) can access an update that is not intended for it	MITM (intercept packets, edit packets, send packets) Cryptographic/signing ability	Director Target Key (1) and Majority of Image Targets keys (ex. 3/4) ECU image encryption/decryption keys if exist and different		4
	Rollback Attack	attacker can send out a previously deployed update to an ECU when a newer update exists	MITM (Intercept/ record packets, edit packets, send packets) Cryptographic/signing ability	Director Target Key (1) and Majority of Image Targets keys (ex. 3/4)	A previously recorded update would have to be renamed to have the next version release counter up,	4
	Mixed Bundle Attack	force an ECU to install incompatible versions of software updates that must not be installed at the same time	MITM (Intercept/ record packets, edit packets, send packets) Cryptographic/signing ability	Director Target Key (1) and Majority of Image Targets keys (ex. 3/4) All ECU image encryption keys if encrypted (might be same)	Assuming the incompatible images on image server, otherwise insider attack What would mixed bundle look like for us considering our ECUs?	3
Control ECU/Vehicle	Injection Attack	Attacker can add their own functionality to an update through injecting data into the update package.	MITM (Intercept/ record packets, edit packets, send packets) Cryptographic/signing ability Developer insider/knowledge to generate valid image	Director Target Key (1) and Majority of Image Targets keys (ex. 3/4) ECU image encryption keys if encrypted		2
	Insider Attack	an attacker can send a fully verified malicious update	MITM (Intercept/ record packets, edit packets, send packets) Developer insider to generate valid image	Director server keys (target, snapshot, timestamp), Image server keys/majority (ex. 3/4 targets, 1/2 snapshot/timestamp) ECU image encryption keys if encrypted	Assuming want to do it "on demand" not piggy-backing off an update being sent out (if others follow this case they need all these keys too)	1

Assumptions:

- This is assuming they can read the traffic / get through TLS
- Most of these assume you are modifying an existing update, otherwise you basically need all the keys
    - Because we only have one top level targets (no delegations) it does not really affect the snapshot/timestamp metadata if you can piggyback
- Image server keys might all be stored separately (not online)
- ECU images might be encrypted on individual basis, might not
- Rank of 1 = most difficult

Ownership	Key Category	Key Num, Signing Threshold (not set, TBD)	Location
Director Repository	Dir. Root	8, 5 for signing quorum	Offline
	Dir. Targets	1, 1	Online, Director Server
	Dir. Snapshot	1, 1	Online, Director Server
	Dir. Timestamp	1, 1	Online, Director Server
Image Repository	Img. Root	8, 5	Offline
	Img. Targets	4, 3	Offline
	Img. Snapshot	2, 1	TBD Online / Offline
	Img. Timestamp	2, 1	TBD Online / Offline
ECU	ECU key	1, 1	ECU
ECU + Director (?)	Image Encryption	1, -	TBD
Time Source (?)	Time Source	1, -	TBD

No supplier / delegated targets keys

Kill-Chain

Goal	Threat	Attack	Barriers
Gather information about the Rohirrim	Read updates	Eavesdrop	Encryption (AES-256) protection adhering to FIPS 140-2
Delay the Rohirrim or allies	Deny installation of updates	Denial-of-service	Monotonically increasing version numbers Vehicle installation reports Network IDS
		Partial installation	“All-or-nothing” update strategy Vehicle installation reports Network IDS
		Arbitrary Invalidate	Multiple signature and hash validation
Disable warriors or equipment	Interfere with ECU function	Infinite data	Download length limits
		Incorrect installation	Use of vehicle and hardware identifiers Images encrypted per vehicle
		Rollback	Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
		Mixed Bundle	Old versions removed from servers Sequential updates must be compatible
Steal equipment from Rohirrim or allies	Control ECU/Vehicle	Injection	Multiple signature and hash validation Key quorum required Multiple servers, offline keys
		Insider Attack	
Wipe out the Rohirrim	?	- Physical	

Uptane

Kill-Chain

Goal	Threat	Attack	Barriers
Gather information about the Rohirrim	Read updates	Eavesdrop	Encryption (AES-256) protection adhering to FIPS 140-2
Delay the Rohirrim or	Deny installation of	Denial-of-service	Monotonically increasing version numbers
		Arbitrary Invalidate	Multiple signature and hash validation
Disable warriors or equipment	Interfere with ECU function	Infinite data	Download length limits
		Incorrect installation	Use of vehicle and hardware identifiers Images encrypted per vehicle
		Rollback	Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
		Mixed Bundle	Old versions removed from servers Sequential updates must be compatible
Steal equipment from Rohirrim or allies	Control ECU/Vehicle	Injection	Multiple signature and hash validation Key quorum required Multiple servers, offline keys
		Insider Attack	
Wipe out the Rohirrim	?	- Physical	

Goal	Threat	Attack
Gather information about the Rohirrim	Read updates	Eavesdrop

SYSTEMS
swri.org

Kill-Chain

Uptane

Goal	Threat	Attack	Barriers
Gather information about the Rohirrim	Read updates	Eavesdrop	Encryption (AES-256) protection adhering to FIPS 140-2
Delay the Rohirrim or	Deny installation of	Denial-of-service	Monotonically increasing version numbers
		Arbitrary Invalidate	Multiple signature and hash validation
Disable warriors or equipment	Interfere with E function		Download length limits
		on	Use of vehicle and hardware identifiers Images encrypted per vehicle
			Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
		Mixed Bundle	Old versions removed from servers ust be compatible
Steal equipment Rohirrim or allies			hash validation e keys
Wipe out the Roh			

Goal	Threat	Attack
Gather information about the Rohirrim	Read updates	Eavesdrop

CRASH JCTD

Barriers
Encryption (AES-256) protection adhering to FIPS 140-2

Kill-Chain

Goal	Threat	Attack	Barriers
Gathe about			
Delay allies			
Disabl equipment	function	In	Use of vehicle and hardware identifiers Images encrypted per vehicle
		R	Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
			ions removed from servers ial updates must be compatible
Steal equipment from Rohirrim or allies	Control ECU/Ve		e signature and hash validation rum required e servers, offline keys
Wipe out the Rohirrim	?		

Delay the Rohirrim or allies	Deny installation of updates	Denial-of-service	Monotonically increasing version numbers Vehicle installation reports Network IDS
		Partial installation	“All-or-nothing” update strategy Vehicle installation reports Network IDS
		Arbitrary Invalidate	Multiple signature and hash validation

CRASH
JCTD

Kill-Chain

Disable warriors or equipment	Interfere with ECU function	Infinite data
		Incorrect installation
		Rollback
		Mixed Bundle

Goal				
Gathe about				protection adhering
Delay allies				sing version numbers ports
				te strategy ports
				d hash validation
Disable warriors or equipment	Interfere with ECU function	In		Download length limits
		In		Use of vehicle and hardware identifiers Images encrypted per vehicle
		Ro		Monotonically increasing version numbers Old versions removed from servers against latest trusted metadata
				ns removed from servers l updates must be compatible
Steal equipment from Rohirrim or allies	Control ECU/Vehic			ignature and hash validation m required ervers, offline keys
Wipe out the Rohirrim	?			

Download length limits
Use of vehicle and hardware identifiers Images encrypted per vehicle
Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
Old versions removed from servers Sequential updates must be compatible

Kill-Chain

Goal	Threat	Attack	Barriers
Gather information about the Rohirrim	Read updates	Eavesdrop	Encryption (AES-256) protection adhering to FIPS 140-2
Delay the Rohirrim or allies	Deny installation of	Denial-of-service	Monotonically increasing version numbers
			Network IDS
			Multiple signature and hash validation
Disable warriors or equipment	Interfere with ECU function		Download length limits
			Use of vehicle and hardware identifiers Images encrypted per vehicle
			Monotonically increasing version numbers Old versions removed from servers Compared against latest trusted metadata
		Mixed Bundle	Old versions removed from servers be compatible
Steal equipment f Rohirrim or allies			ash validation keys
Wipe out the Rohi			

Steal equipment from Rohirrim or allies	Control ECU/Vehicle	Injection Insider Attack

Multiple signature and hash validation
Key quorum required
Multiple servers, offline keys

Metrics

Proposal – create benchmark metrics for implementations

- What metrics would be valuable?
  - Example sizes for resource requirements
    - Separate primary and secondary
      - Minor value for server instances
    - Memory
    - CPU
    - Disk
  - Speed of execution for specific functions
  - Bandwidth
- Can metrics be gathered given the different HW?
  - FPGA, Embedded, Microkernel, ASIC