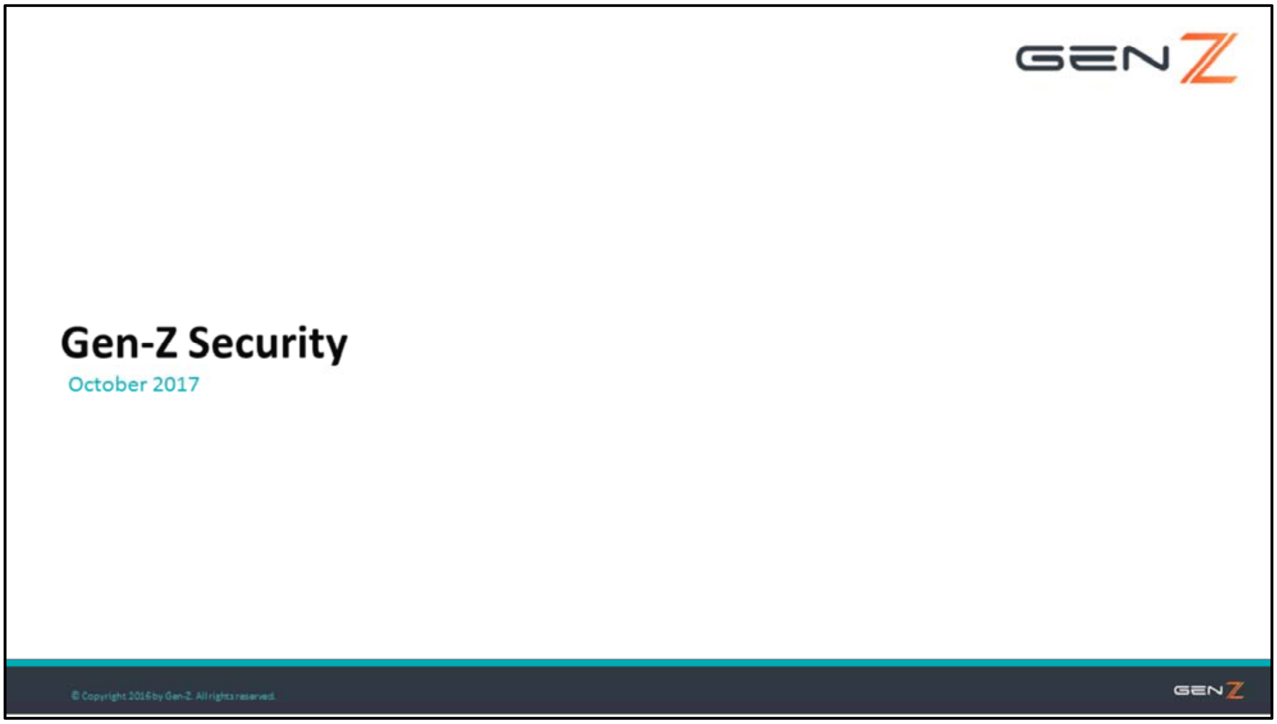
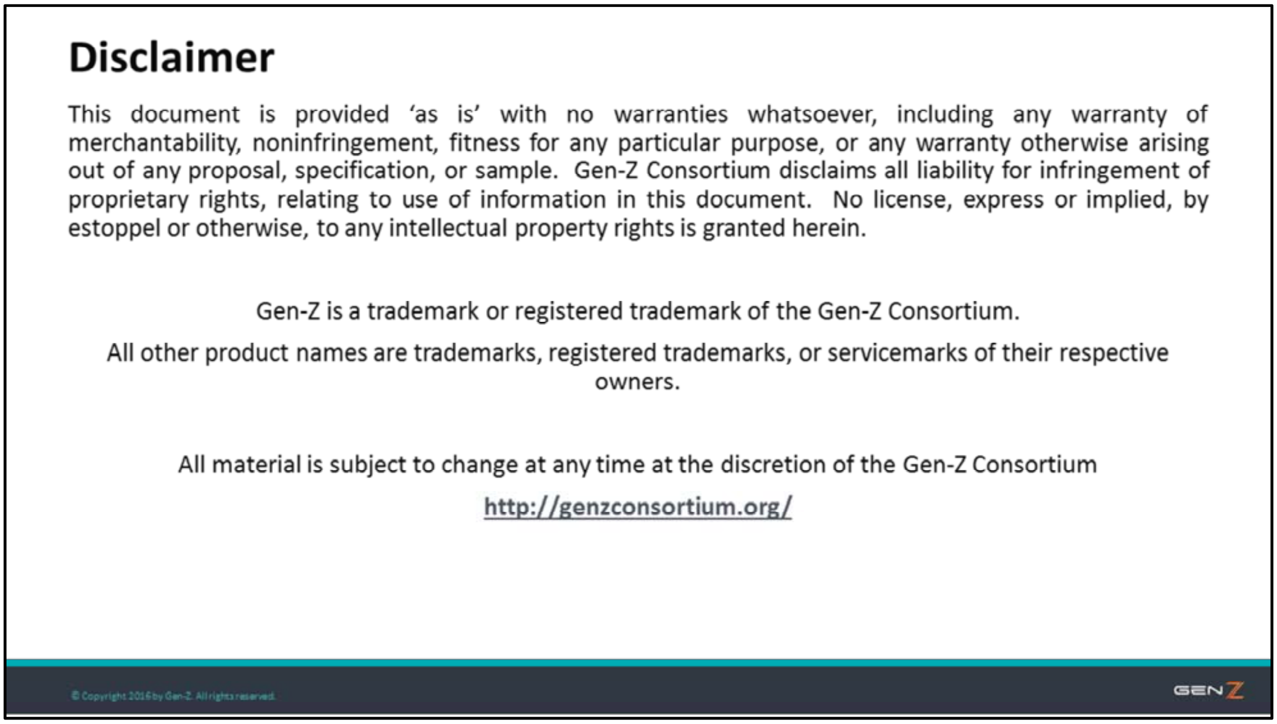


# Gen-Z Security

October 2017

This presentation covers Gen-Z's Security capabilities.



# Disclaimer

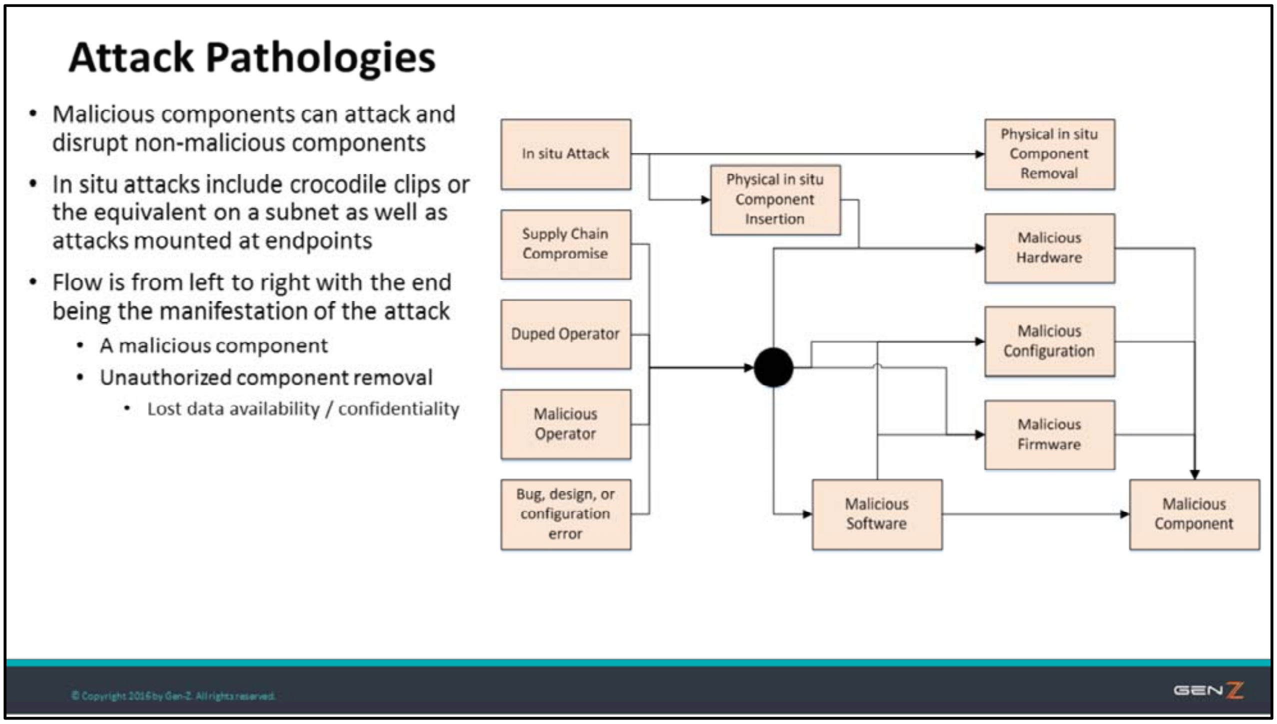
This document is provided 'as is' with no warranties whatsoever, including any warranty of merchantability, noninfringement, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. Gen-Z Consortium disclaims all liability for infringement of proprietary rights, relating to use of information in this document. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Gen-Z is a trademark or registered trademark of the Gen-Z Consortium.

All other product names are trademarks, registered trademarks, or servicemarks of their respective owners.

All material is subject to change at any time at the discretion of the Gen-Z Consortium

http://genzconsortium.org/



## Attack Pathologies

- Malicious components can attack and disrupt non-malicious components
- In situ attacks include crocodile clips or the equivalent on a subnet as well as attacks mounted at endpoints
- Flow is from left to right with the end being the manifestation of the attack
  - A malicious component
  - Unauthorized component removal
    - Lost data availability / confidentiality

In situ Attack

Supply Chain Compromise

Duped Operator

Malicious Operator

Bug, design, or configuration error

Physical in situ Component Insertion

Physical in situ Component Removal

Malicious Hardware

Malicious Configuration

Malicious Firmware

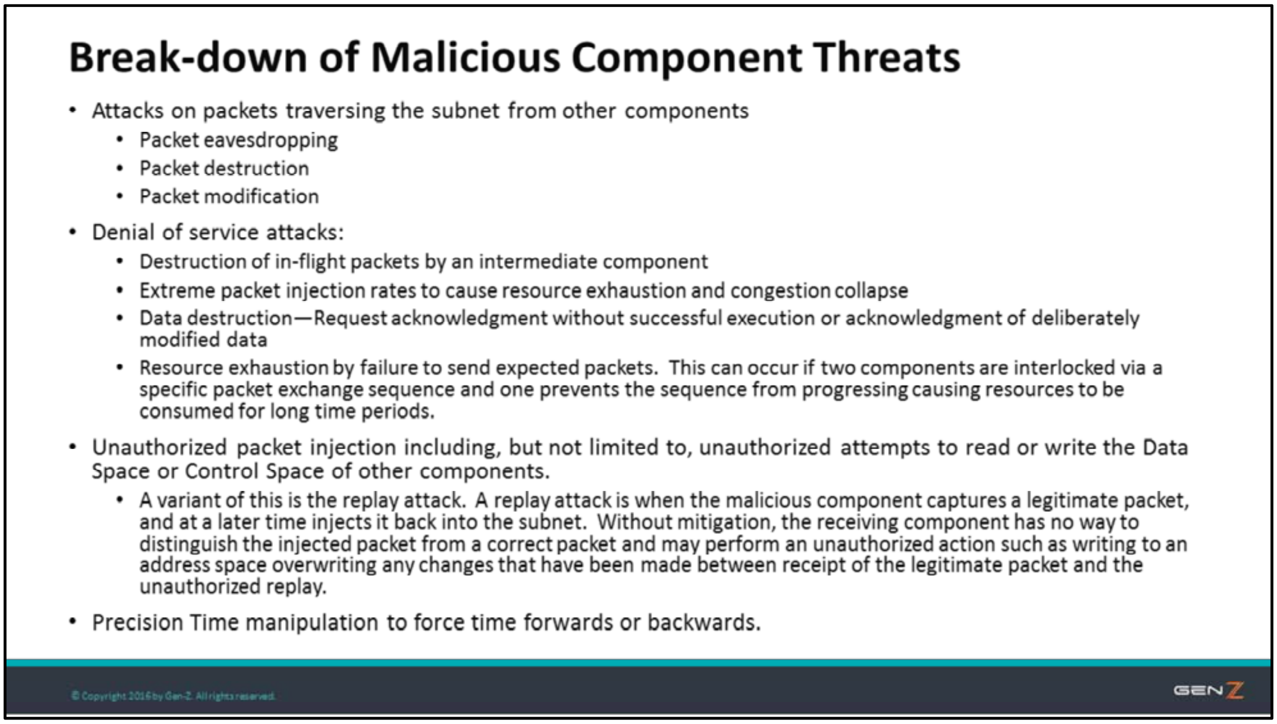
Malicious Software

Malicious Component

GEN Z

Gen-Z architecture assumes every component is an attack vector.  This is critical to appreciate, as time and again cyber attacks have exploited hardware and software that take no steps to authenticate access and establish trust.

This slide illustrates how attacks are carried out today at all levels irrespective of the interconnect used.

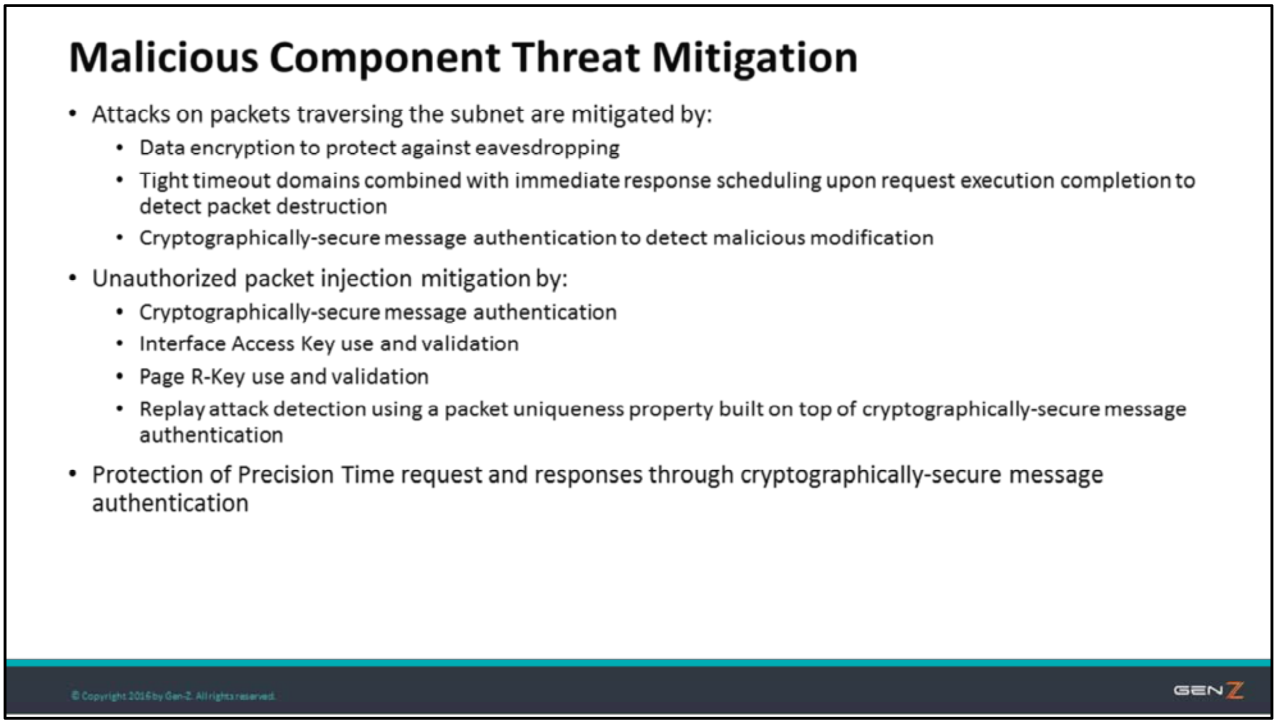


## Break-down of Malicious Component Threats

- Attacks on packets traversing the subnet from other components
    - Packet eavesdropping
    - Packet destruction
    - Packet modification
- Denial of service attacks:
    - Destruction of in-flight packets by an intermediate component
    - Extreme packet injection rates to cause resource exhaustion and congestion collapse
    - Data destruction—Request acknowledgment without successful execution or acknowledgment of deliberately modified data
    - Resource exhaustion by failure to send expected packets. This can occur if two components are interlocked via a specific packet exchange sequence and one prevents the sequence from progressing causing resources to be consumed for long time periods.
- Unauthorized packet injection including, but not limited to, unauthorized attempts to read or write the Data Space or Control Space of other components.
    - A variant of this is the replay attack. A replay attack is when the malicious component captures a legitimate packet, and at a later time injects it back into the subnet. Without mitigation, the receiving component has no way to distinguish the injected packet from a correct packet and may perform an unauthorized action such as writing to an address space overwriting any changes that have been made between receipt of the legitimate packet and the unauthorized replay.
- Precision Time manipulation to force time forwards or backwards.

GEN Z

This slide provides a high-level breakdown on malicious component threats. Again, these threats are interconnect independent, and cyber attacks make use of these on a regular basis.

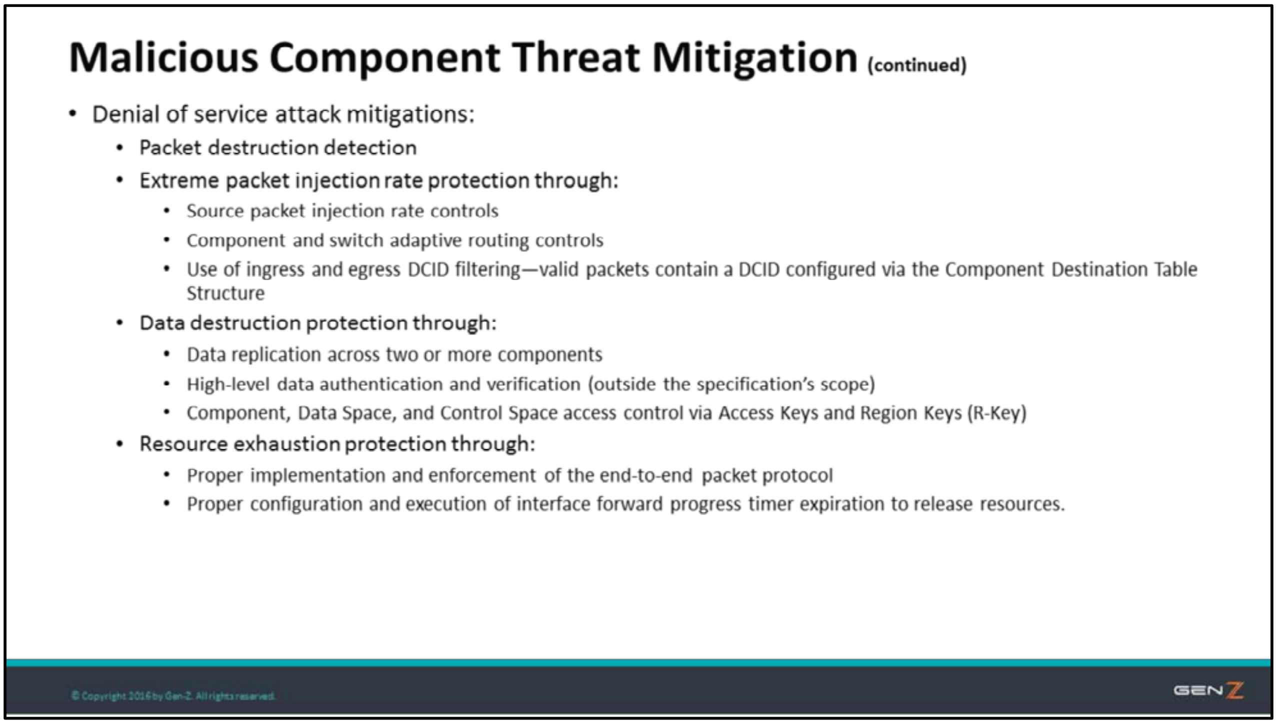


## Malicious Component Threat Mitigation

- Attacks on packets traversing the subnet are mitigated by:
  - Data encryption to protect against eavesdropping
  - Tight timeout domains combined with immediate response scheduling upon request execution completion to detect packet destruction
  - Cryptographically-secure message authentication to detect malicious modification
- Unauthorized packet injection mitigation by:
  - Cryptographically-secure message authentication
  - Interface Access Key use and validation
  - Page R-Key use and validation
  - Replay attack detection using a packet uniqueness property built on top of cryptographically-secure message authentication
- Protection of Precision Time request and responses through cryptographically-secure message authentication

This slide describes a set of high-level actions that can be used to mitigate the threats posed by malicious components.  All of these actions are supported by Gen-Z, and can be applied to any Gen-Z-based solution.

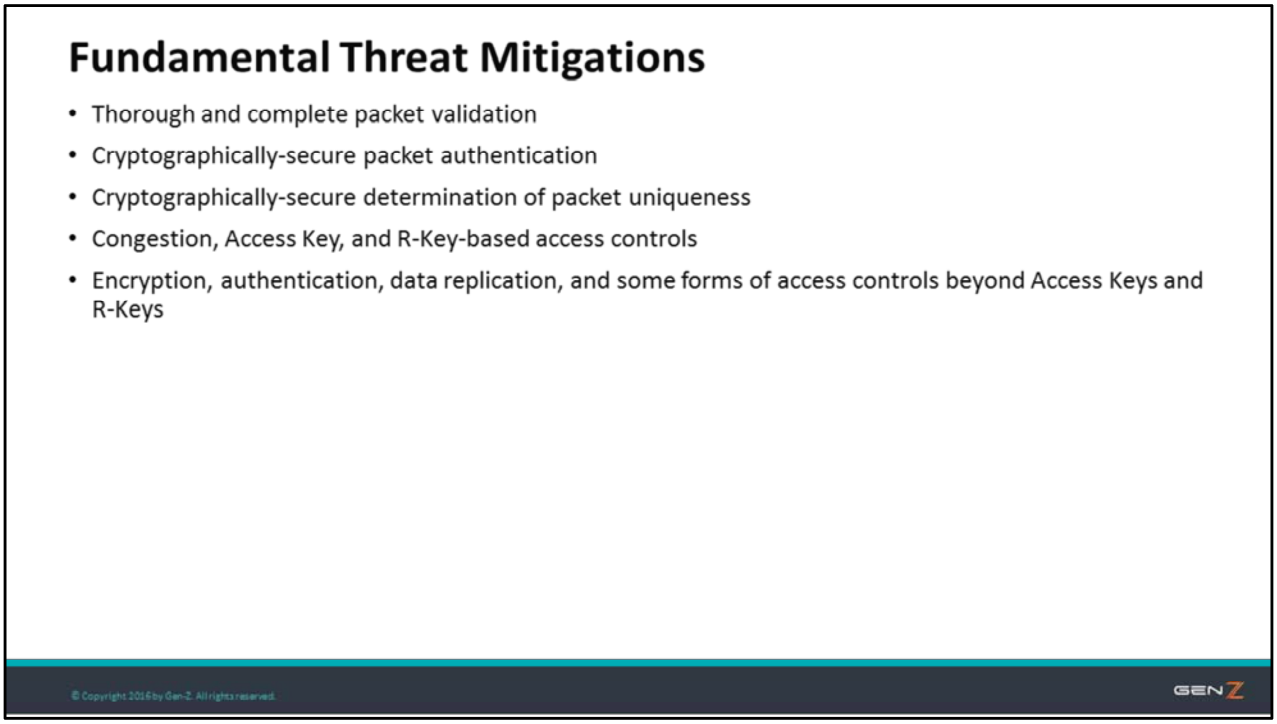
- The payload within a Gen-Z packet can be encrypted to ensure privacy.
- Gen-Z supports packet deadlines to enable more aggressive end-to-end request packet retransmission timers.
- Gen-Z supports Hash-based Message Authentication (HMAC) that can be included in any explicit OpClass packet.
- Gen-Z Explicit OpClass packets contain Access Keys and select request packets contain R-Keys to provide hardware-enforced access control.  Further, Gen-Z supports additional access control and packet filtering to ensure components can communicate only with configured peers.
- Gen-Z explicit OpClass packets can include an anti-replay field (sequence number or precision time-based) to ensure request packets are not intercepted and replayed multiple times.
- Precision time request and responses are exchanged using explicit OpClass packets, and can be cryptographically secured.



## Malicious Component Threat Mitigation (continued)

- Denial of service attack mitigations:
  - Packet destruction detection
  - Extreme packet injection rate protection through:
    - Source packet injection rate controls
    - Component and switch adaptive routing controls
    - Use of ingress and egress DCID filtering—valid packets contain a DCID configured via the Component Destination Table Structure
  - Data destruction protection through:
    - Data replication across two or more components
    - High-level data authentication and verification (outside the specification's scope)
    - Component, Data Space, and Control Space access control via Access Keys and Region Keys (R-Key)
  - Resource exhaustion protection through:
    - Proper implementation and enforcement of the end-to-end packet protocol
    - Proper configuration and execution of interface forward progress timer expiration to release resources.

Gen-Z architecture specifies mechanisms that cover all of the listed mitigations.



## Fundamental Threat Mitigations

- Thorough and complete packet validation
- Cryptographically-secure packet authentication
- Cryptographically-secure determination of packet uniqueness
- Congestion, Access Key, and R-Key-based access controls
- Encryption, authentication, data replication, and some forms of access controls beyond Access Keys and R-Keys

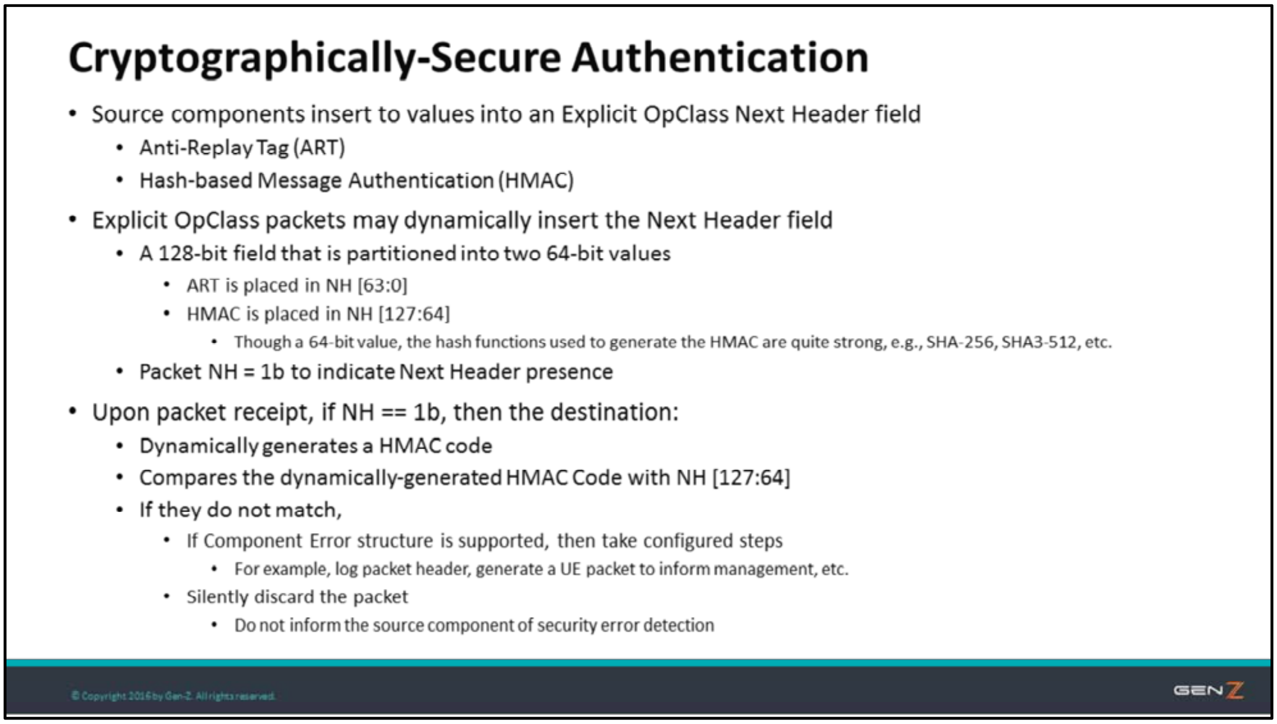
Gen-Z specifies the detailed steps to validate all packets and the actions to take upon detecting an issue.

Gen-Z supports HMAC to provide strong packet authentication.

Gen-Z supports anti-replay protection in conjunction with HMAC to ensure packet uniqueness. Further, each packet is uniquely identified by a combination of its source component identifier, destination component identifier, and Tag or sequence number (SOD-only).

Gen-Z supports multiple packet injection rate (congestion management) and access control mechanisms to provide hardware-enforced isolation and permission validation.

Components can encrypt data payloads to ensure privacy, can apply higher-layer authentication mechanisms, e.g., service or service instance information, can replicate data or interleave memory in conjunction with erasure codes, etc.



## Cryptographically-Secure Authentication

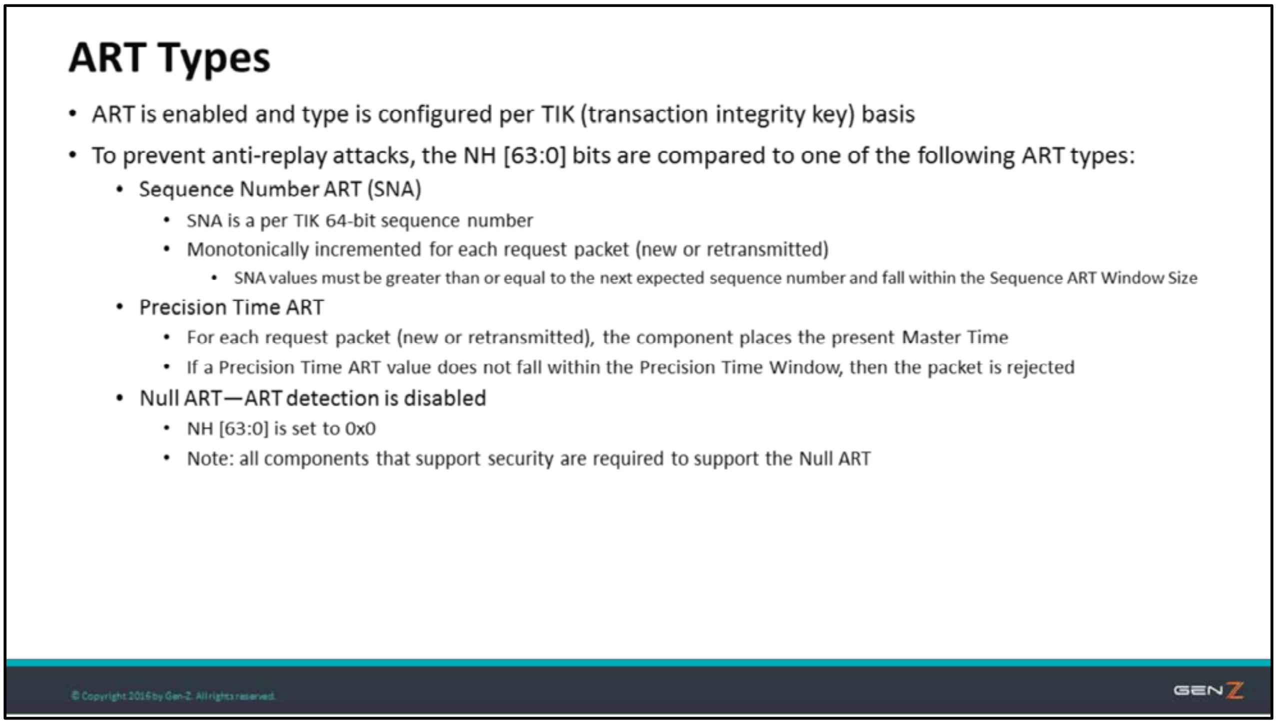
- Source components insert to values into an Explicit OpClass Next Header field
  - Anti-Replay Tag (ART)
  - Hash-based Message Authentication (HMAC)
- Explicit OpClass packets may dynamically insert the Next Header field
  - A 128-bit field that is partitioned into two 64-bit values
    - ART is placed in NH [63:0]
    - HMAC is placed in NH [127:64]
      - Though a 64-bit value, the hash functions used to generate the HMAC are quite strong, e.g., SHA-256, SHA3-512, etc.
  - Packet NH = 1b to indicate Next Header presence
- Upon packet receipt, if NH == 1b, then the destination:
  - Dynamically generates a HMAC code
  - Compares the dynamically-generated HMAC Code with NH [127:64]
  - If they do not match,
    - If Component Error structure is supported, then take configured steps
      - For example, log packet header, generate a UE packet to inform management, etc.
    - Silently discard the packet
      - Do not inform the source component of security error detection

All explicit OpClass packets may contain the Next Header field (the field is present if NH == 1b.

Gen-Z specifies control structures that enable the Next Header field to be applied to all explicit OpClass communications, to only control plane communications, on a per peer component basis, etc.

Gen-Z specifies the Component Security structure to provide security certificate and TIK (Transaction Integrity Key) management. A component can support multiple certificates and TIKs enabling different authentication to be applied per peer component.

If a component detects a security violation, it can be configured to immediately inform management without informing the source component. Management can take steps to isolate, reset, or shutdown the source component based on customer-specific policies.

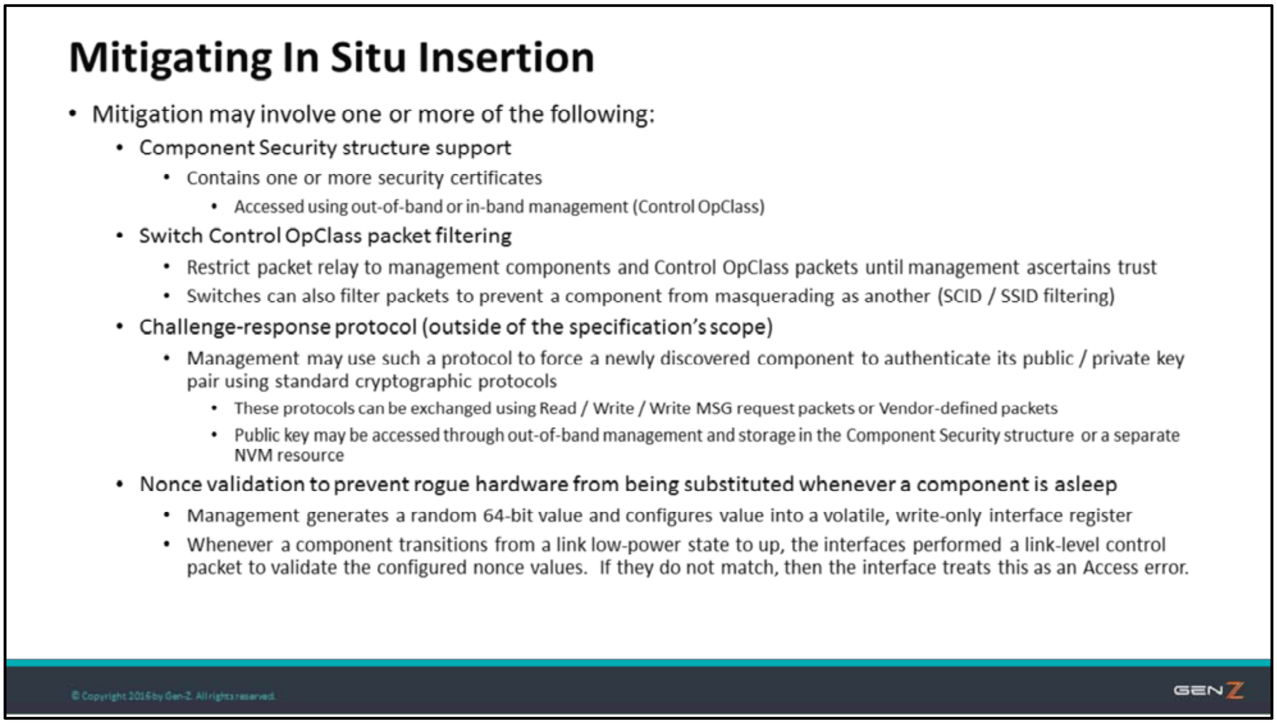


## ART Types

- ART is enabled and type is configured per TIK (transaction integrity key) basis
- To prevent anti-replay attacks, the NH [63:0] bits are compared to one of the following ART types:
  - Sequence Number ART (SNA)
    - SNA is a per TIK 64-bit sequence number
    - Monotonically incremented for each request packet (new or retransmitted)
      - SNA values must be greater than or equal to the next expected sequence number and fall within the Sequence ART Window Size
  - Precision Time ART
    - For each request packet (new or retransmitted), the component places the present Master Time
    - If a Precision Time ART value does not fall within the Precision Time Window, then the packet is rejected
  - Null ART—ART detection is disabled
    - NH [63:0] is set to 0x0
    - Note: all components that support security are required to support the Null ART

The current specification supports three ART types:

- A per TIK sequence number.   New or retransmitted packets must contain a SNA that is greater than or equal to the expected sequence number.  Further, the SNA needs to fall within a window to prevent a malicious component from advancing the sequence number such that it will be easier to hijack communications.
- A Precision time ART requires global precision time management, i.e., not on a per TIK basis.  All participating components advance their understanding of precision time, hence, any component can detect if a new packet falls within the configured precision time window.  Packets that fall outside of this window are treated as malicious.
- Null ART—this indicates ATR detection is disabled.



## Mitigating In Situ Insertion

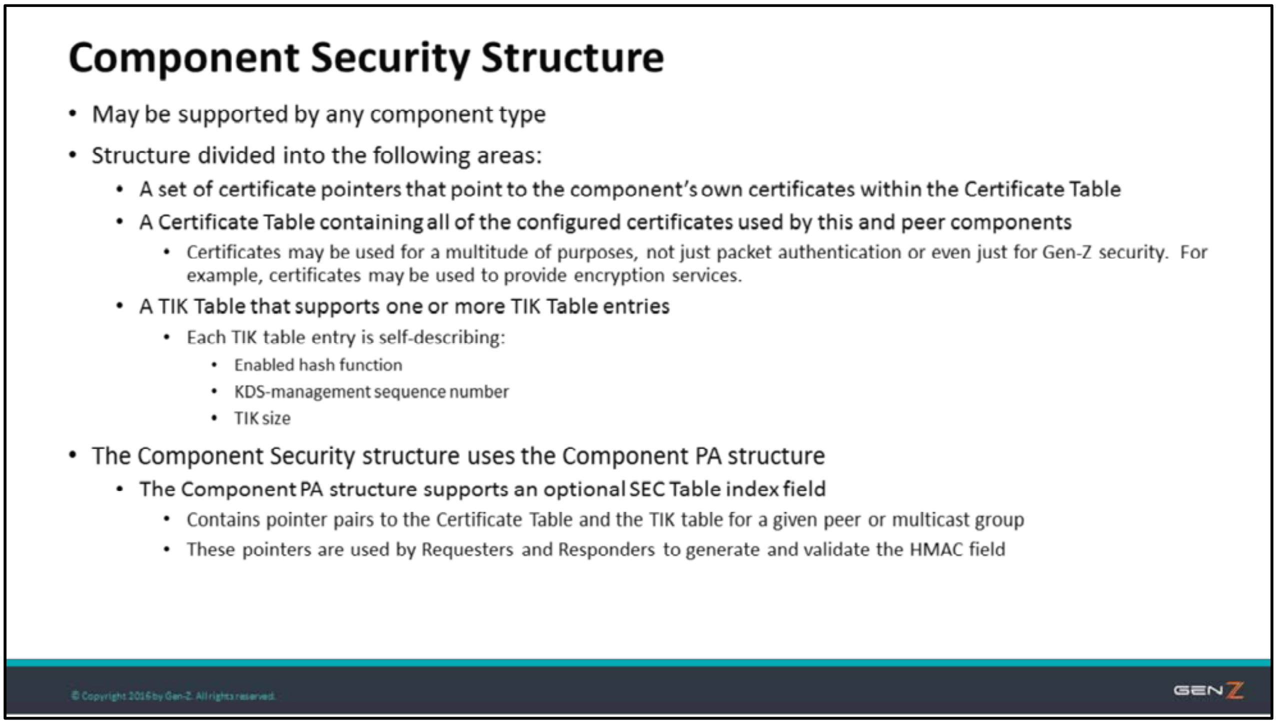
- Mitigation may involve one or more of the following:
  - Component Security structure support
    - Contains one or more security certificates
      - Accessed using out-of-band or in-band management (Control OpClass)
  - Switch Control OpClass packet filtering
    - Restrict packet relay to management components and Control OpClass packets until management ascertains trust
    - Switches can also filter packets to prevent a component from masquerading as another (SCID / SSID filtering)
  - Challenge-response protocol (outside of the specification's scope)
    - Management may use such a protocol to force a newly discovered component to authenticate its public / private key pair using standard cryptographic protocols
      - These protocols can be exchanged using Read / Write / Write MSG request packets or Vendor-defined packets
      - Public key may be accessed through out-of-band management and storage in the Component Security structure or a separate NVM resource
  - Nonce validation to prevent rogue hardware from being substituted whenever a component is asleep
    - Management generates a random 64-bit value and configures value into a volatile, write-only interface register
    - Whenever a component transitions from a link low-power state to up, the interfaces performed a link-level control packet to validate the configured nonce values. If they do not match, then the interface treats this as an Access error.

In Situ insertion is the physical interposition of hardware, e.g., crocodile clips or a Y-cable to physically collect and / or modify packets.

If all components support the Component Security structure, then a key management system can be used to populate the security certificates and TIKs. Certificates and TIKs can be populated such that these values can be trusted. Management ensures that only authorized components can communicate with one another.

Switches can provide a limited level of packet filtering. For example, during leaf component configuration, a switch can filter any non-Control OpClass packets to prevent the component or any other component from communicating with the component under configuration. Once configured or through a Link CTL exchange, the switch can filter the SCID / SSID in all packets to ensure that a leaf component only transmits packets corresponding to its component identifier.

Though outside of the specification's scope, management can use a challenge-response protocol to authenticate the public / private keys. Such a protocol can be transported in Gen-Z Read / Write or Write MSG or Vendor-defined packets.



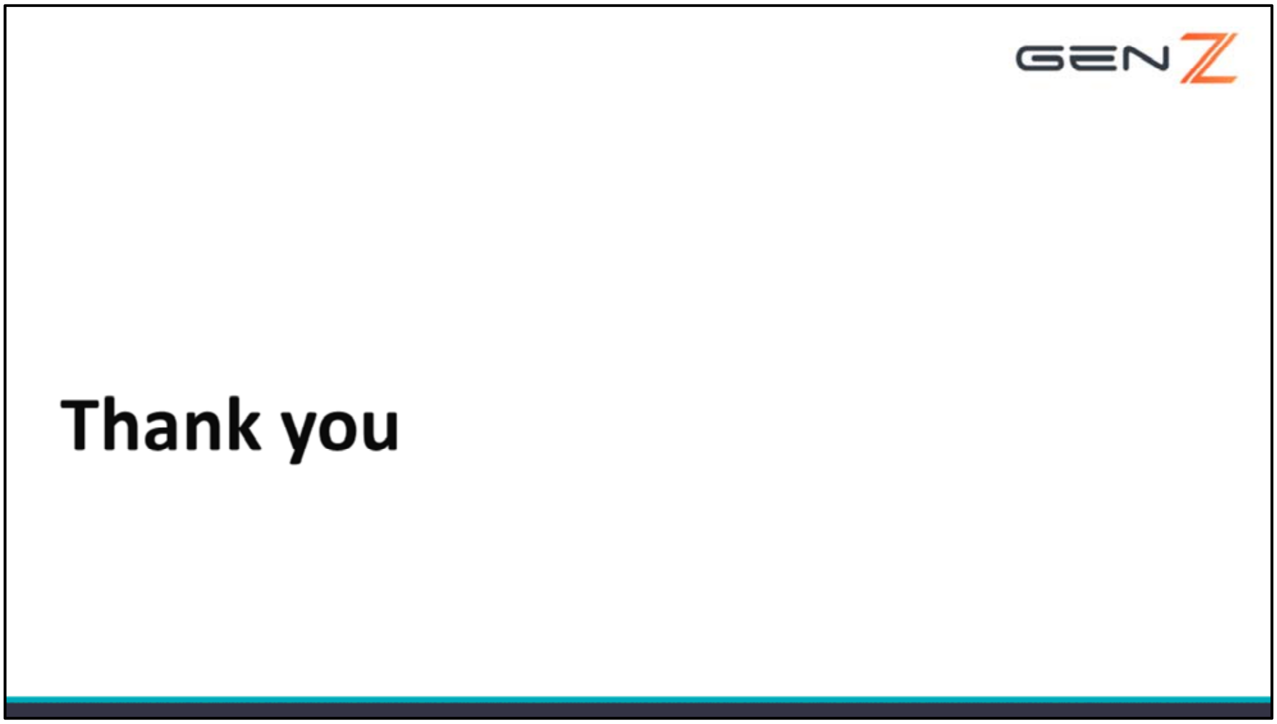
## Component Security Structure

- May be supported by any component type
- Structure divided into the following areas:
  - A set of certificate pointers that point to the component's own certificates within the Certificate Table
  - A Certificate Table containing all of the configured certificates used by this and peer components
    - Certificates may be used for a multitude of purposes, not just packet authentication or even just for Gen-Z security. For example, certificates may be used to provide encryption services.
  - A TIK Table that supports one or more TIK Table entries
    - Each TIK table entry is self-describing:
      - Enabled hash function
      - KDS-management sequence number
      - TIK size
- The Component Security structure uses the Component PA structure
  - The Component PA structure supports an optional SEC Table index field
    - Contains pointer pairs to the Certificate Table and the TIK table for a given peer or multicast group
    - These pointers are used by Requesters and Responders to generate and validate the HMAC field

The Component Security structure is used to manage security services. It contains a set of tables:

- Pointers to the component's security certificates. A component can support multiple security certificates.
- A certificate table containing all of the security certificates used by this components and its peer. This table can contain security certificates that are not used for Gen-Z, e.g., for higher-level security services such as encryption.
- A Transaction Integrity Key table. A TIK is a "secret" shared between the communicating components. Each entry specifies one of the 10 possible hash functions used perform HMAC, a sequence number to track how often this entry was updated, and the size of the TIK itself.

A component uses the Component PA structure to locate the Certificate and TIK table entries. This enables a component to quickly identify a unique entry per component or to use a wildcard set of values applied to all communications.



# Thank you

This concludes this presentation.  Thank you.