WHITE PAPER

Security Architecture of Pensando's Distributed Services Platform

Introduction

A fundamental aspect of Pensando's Distributed Services Platform is that critical infrastructure security functions can be run in an independent, isolated environment apart from the server. This isolation provides a significant boost to data center security – but only if the hardware architecture and software implementation have a sound basis in secure systems design and follow secure development principles. For an independent security and policy enforcement system to be effective, it must itself be secure since it will likely be the focus of attacker.

Goals of this White Paper

- Describe the specific security features of Pensando's Distributed Services Card
- Discuss the types of attacks that Pensando's security elements can protect against

Pensando's Security Centric Solution

Hardware-Enforced Security

The highest level of security assurance is best achieved with isolated hardware that is architected to be secure from all expected threats, both internal and external. Pensando has developed the ***Capri*** Programmable Services Processor (PSP), based on a domain-specific architecture that integrates a wide range of security features described in this document. This device is deployed on Pensando's Distributed Services Cards (**DSCs**) that are installed in data center servers to bring secure services directly to the server edge.

Comprehensive "Security-First" Approach

Pensando understands that comprehensive security must cover the entire lifecycle of a product – not just the design elements, but also manufacturing, deployment, upgrades and decommissioning. For this reason, Pensando has developed a security infrastructure that implements a chain of trust extending to all the hardware and software components of the solution. The Pensando Distributed Services Platform utilizes Public Key technology and Certificate Authorities as a cornerstone of this approach, mandating that each component prove its identity and integrity to the central manager.

Pensando Certificate Authority Chain

Digital Certificates create a binding between entities and public keys, enables other entities to verify those public key bindings, and provide the services needed for ongoing management of keys in a distributed system. A primary purpose of the Pensando CA chain is to issue digital certificates to Pensando hardware and software components as part of the hardware manufacturing and software release processes.

Pensando's CA hierarchy is 2-levels deep. There is a single, Pensando Systems Root CA at the top of the hierarchy. One level below the root, there are issuing CAs for specific functions.

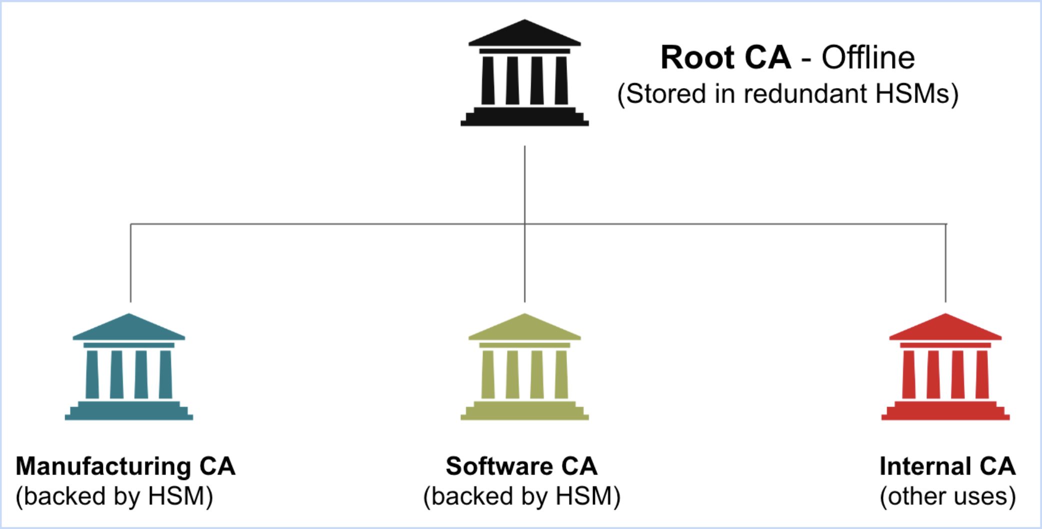


Fig 1. Pensando CA hierarchy

The private key of the Root CA is kept offline in redundant commercial Hardware Security Modules (HSMs). Private keys for the intermediate CAs are also generated and stored in separate Hardware Security Modules (HSMs) that are accessed via Manufacturing/Operations and the Software Release organization. The access to tasks such as certificate issuance and image signing is precisely controlled and authenticated and access is enforced over encrypted VPN connections. All operations are logged at the server and HSM level.

Use Cases

The use-cases for the Pensando certificate hierarchy include:

- **Hardware attestation**: Each chip manufactured by Pensando has an embedded unique identity in the form of a private key (also known as "Endorsement Key" or "EK") and certificate. The EK and the certificate are burned-in at manufacturing time and cannot be modified afterwards. The certificate contains the serial number of the chip and is signed by the Pensando CA. The EK can be used to sign Attestation Certificates, but cannot be extracted from the chip.
- **Secure boot**: The Pensando Distributed Services Card (**DSC**) boot sequence consists of several steps involving multiple software images. All images are signed and each image checks the signature of the next image before being loaded. If the verification fails, the normal boot process stops and the card goes into a recovery mode. If the boot process completes successfully, the card is guaranteed to run genuine, unaltered Pensando software. This pre-condition is verified before admitting the card to the cluster.
- **Software validation**: Pensando's Policy and Services Manager (**PSM**) verifies the signature of target images before performing any software upgrade.
- **Connection validation**: All connections between the PSM and DSCs are protected with TLS and use bi-directional authentication based on digital certificates. DSCs will not be admitted into use until they have been authenticated.

The combination of these mechanisms protects the Pensando Distributed Services Platform against rogue hardware or software components attempting to maliciously insert themselves into the system.

Hardware Security from the Ground Up

Pensando's cryptography hardware employs a comprehensive suite of security features starting with secure boot, key management, secure key store and enabling high-performance security protocols such as TLS/DTLS, IPsec and AES-XTS data-at-rest encryption. The security capabilities are implemented in two distinct subsystems within Pensando's Programmable Services Processors:

- *PenTrust*™ subsystem – Root of Trust, Secure boot, key management and control plane security
- *PenAccel*™ subsystem – High-performance accelerators for the data path, including crypto engines

PenTrust Subsystem

Internal to Pensando's Capri silicon at the heart of each Distributed Services Card (DSC) is a security element called the PenTrust Subsystem that is responsible for the fundamental protection mechanisms in the device. The module is isolated from the rest of the chip and has its own CPU, ROM, RAM, and cryptographic engines, similar to Trusted Platform Modules (TPM) used in server systems. *PenTrust* accesses chip resources outside of its secure perimeter via its own bus mastering DMA engine, and it can receive requests (e.g. to create keys or sign certificates) via a narrow, secure-access interface. Other modules can deposit requests and asynchronously pick up responses as they become available.

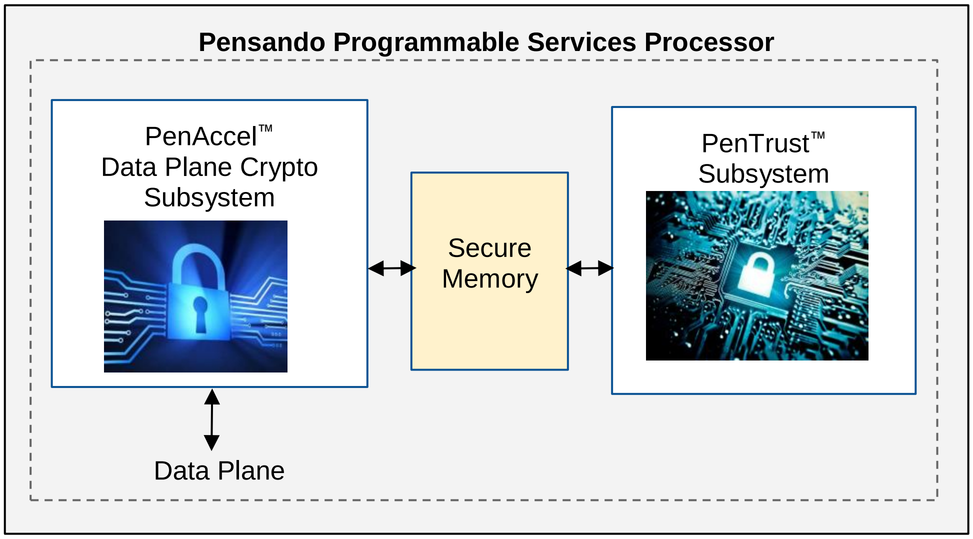

Pensando Programmable Services Processor
PenAccel™ Data Plane Crypto Subsystem
Secure Memory
PenTrust™ Subsystem
Data Plane

Fig 2. Capri P4 Programmable Processor

The *PenTrust* subsystem provides trusted crypto services such as asymmetric (public key) sign and verify operations, as well as symmetric (secret key) encryption/decryption and secure hashing operations. It also enables secure key store and management via primitives to generate keys, wrap/unwrap keys, and import/export them in encrypted form.

Root of Trust

The *PenTrust* subsystem is the Root-of-Trust (RoT) for the Pensando Programmable Services Processor. As discussed in the Secure Boot section below, *PenTrust* is the first subsystem to boot at power-on, and it does so from its immutable embedded ROM; making it the first link in the secure boot chain. At the heart of the RoT is a Physically Unclonable Function (PUF). This is a specialized silicon element, tied to specific physical properties of each silicon die, that provides a device-specific seed key. The result is that each chip produced by Pensando will have a unique PUF seed key that cannot be read outside of the device nor tampered with. The seed key is exclusively used to derive a 256-bit AES Storage Root Key (SRK) and an ECDSA-P384 Endorsement Key (EK).

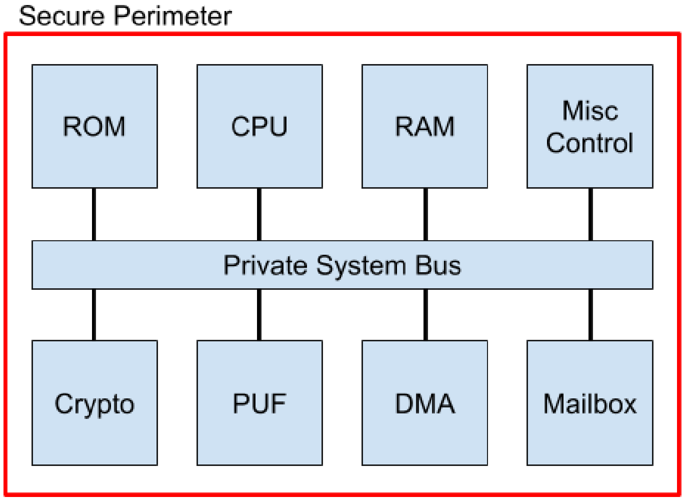


Fig 3. PenTrust Subsystem

The **SRK** is used to encrypt (wrap) secrets (such as sensitive configuration, symmetric encryption keys, and private asymmetric keys) that can then be written to non-volatile storage, or exported off of the device for remote storage. As the SRK never leaves the RoT secure perimeter, only the origin RoT can decrypt and recover the data.

The **EK** is used to build a chain of trust that can attest the authenticity of a DSC device. During manufacturing a Pensando-signed X.509 certificate is installed into the device which authenticates the public part of the EK. The private part of the EK never leaves the RoT. The RoT generates a 2nd-level Attestation Key-pair (authenticated via an EK-signed X.509 certificate), and that Attestation Key can be used to sign Nonce challenges. The Nonce challenge, along with the certificate chain rooted at the Pensando public Root Certificate, allows an external entity, such as Pensando's Policy and Services Manager (PSM), to authenticate the DSC as genuine.

Secure Boot

Secure Boot technology ensures that only software from a trusted source is allowed to run on the Pensando Distributed Services Card.

At device power-on / reset, the *PenTrust* RoT processor begins executing from its immutable trusted program ROM, part of the ASIC die. The main ARM application processors are held in reset while the RoT performs its self-tests and initialization.

The RoT derives its SRK from the PUF and fetches its Critical Security Parameters (CSPs) from read-only flash. These SRK-encrypted parameters include both RoT firmware and main application software authentication and decryption keys. As noted above, the use of the SRK ensures that the CSPs cannot be forged even by direct physical flash attack. The RoT loads its runtime firmware from flash into its secure, isolated memory, authenticates and decrypts it using keys from the CSP, and then passes control to it. Two Flash-based images are supported for redundancy / upgrade of the *PenTrust* firmware.

The RoT then proceeds to load the main ARM application processor's first-stage boot block into secure memory that is external to the RoT (accessible to the application processor). The application processor's authentication and decryption keys are used to verify the application processor's image, and if successful the primary ARM core is taken out of reset and begins executing its trusted, authenticated code.

The primary ARM application processor boot sequence follows the ARM Trusted Firmware boot flow, with a multi-step sequence bringing the system up from first-stage boot to fully operational, running Linux with an authenticated filesystem image. At each step, newly loaded programs, keys, or data are authenticated by the previous step, creating an unbroken chain-of-trust back to the RoT ROM.

Dual full-system images are provided for redundancy / upgrade, and a read-only Golden Image provides a recovery option should both main images fail.

HardGap™ Firewall from Host

In order to provide isolation from server-based vulnerabilities including deliberate attack or a rogue process writing through memory, reliable protection of host PCIe access to the DSC memory regions is required. Pensando has developed a unique hardware-based firewall at the PCIe-memory interface inside the Programmable Services Processor.

Unlike traditional PCIe-attached devices, Pensando Programmable Services Processors provide no intrinsic device access to the host complex. In other words, the host has no de-facto privilege to control the device. The PCIe device tree that the host system sees is established entirely by DSC trusted software, and enforced by hardware PCIe transaction filters. The DSC software controls the PCIe devices presented by the DSC to the host, as well as the individual resource BARs made available to drivers. All controls are software-defined, hardware enforced, and compliant with the PCI Express standards. Together, these controls allow Pensando devices to present a number of different device, management, and security profiles to the host, while maintaining strict protection against unauthorized access.

Secure Memory Regions

Pensando Programmable Services Processors implement the ARM TrustZone® architecture, providing for Secure/Trusted and Normal application memory spaces. The support is implemented throughout the device, and covers hardware DMA engines, DRAM memory (HBM), and even individual memory-mapped hardware registers.

Memory regions can be configured as *secure*, with secure memory regions being accessible only by secure hardware components. Secure components include:

- ARM software running at a secure level (e.g. EL3, S-EL1)
- The *PenTrust* RoT
- The *PenAccel* Data Plane Crypto DMA

Secure memory access is enforced in hardware. Specifically, every internal bus master has its memory transactions policed by a security filter. For memory ranges declared as secure, only certain masters are permitted access, and then only if their transactions are marked as secure (allowing a trusted master to be specific about individual transactions). The security filters themselves, along with other registers that affect security, are *intrinsically secure*; meaning they require secure access. In practice this means that overall security control is under the purview of ARM software running at a secure level.

The *PenAccel* Crypto DMA supports encryption keys placed in secure memory, out of reach of a non-secure attacker. The high-performance encryption engines can read key material using secure memory transactions, but are restricted to using normal memory transactions to fetch data to be encrypted/decrypted. The configuration is itself secure-only, preventing a non-secure attacker from abusing the system to expose keys as data.

Random Number Generator (RNG) Support

The foundation for any crypto subsystem is a high-quality source of random numbers, having at its core a source of entropy (unpredictability) but also with randomness properties that approximate white noise.

Both of the cryptographic hardware subsystems, *PenTrust* and *PenAccel* contain RNGs that are NIST 800-90A compliant using Hash-DRBG (SHA256) support with security strengths of 112, 128, 192 and 256 bits. The DRBG is seeded by a non-deterministic entropy source block which is a NIST 800-90B compliant oscillator ring-based implementation.

The *PenTrust* RNG is generally used only for random number requirements inside of the secure boundary of that subsystem. The *PenAccel* RNG can provide random numbers to other subsystems within the device, including applications running on the ARM complex.

Symmetric Cryptographic Operations

Both *PenTrust* and *PenAccel* Crypto subsystems support a range of Hash and HMAC algorithms including SHA-1, SHA-2 (SHA224, SHA256, SHA384 and SHA512). They also support various AES variants such as GCM, CCM, and CBC.

The *PenAccel* crypto subsystem also supports AES-XTS as well as the SHA-3 Hash algorithm variants SHA3-224, SHA3-256, SHA3-384 and SHA3-512.

The hardware cryptographic engines in the Pensando DSCs have been tested and certified for compliance through the NIST CAVP program – enablers for overall FIPS 140-2 compliance.

Asymmetric Cryptographic Operations

The PenTrust subsystem security processor supports RSA Encrypt/Decrypt with OAEP and EME-PKCS padding schemes and signature operations with EMSA-PKCS and PSS padding schemes.

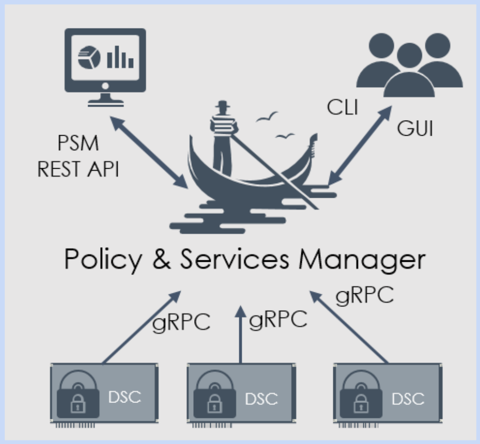
The subsystem also supports DSA and ECDSA signature algorithms.

The high performance PenAccel cryptographic subsystem supports modular arithmetic primitives in addition to RSA (2K, 3K and 4K key sizes), ECC, ECDSA and DSA operations. For Elliptic Curve Cryptography, the supported curves are P-192, P-224, P-256, P-384, P-521 and Curve25519.

End-To-End Trust: Management To Data Plane

The management plane of the Pensando platform comprises the Policy and Services Manager (PSM) and agents running on the DSCs. The PSM in turn consists of multiple services running on controller nodes.

All communications between these software components are encrypted and authenticated using mutual TLS with an enforced minimum TLS version of 1.2. The client and the server authenticate each other using certificates issued by a Certificate Authority embedded in the PSM. Each PSM CA instance has a unique signing key, so the certificates it generates are only valid within the boundaries of a specific PSM installation. Customers can upload their own CA-issued x.509 certificate and private key to the PSM in order to secure the northbound interfaces over REST/HTTPS.



Authenticating DSCs

When a Pensando DSC first tries to join a PSM cluster, it has no cluster-specific credentials yet, so it cannot authenticate itself using TLS. The DSC, however, has an embedded Endorsement Key (EK) and a certificate containing a unique ID issued by the Pensando CA. The PSM verifies the authenticity of the card by giving it a random nonce to sign with the EK and verifying the signature with the public key in the EK certificate. The signature of the EK certificate in turn is verified with the public key(s) issued by the Pensando CAs, up to the root CA. If verification of the DSC identity is successful, the PSM can automatically admit it to the cluster or put it in “pending” state until an administrator admits it. Once the DSC is admitted to the cluster, it is given client certificates by the cluster CA so it can establish communications with the other cluster components. The DSC stores the identity of the cluster it joined and refuses to connect to any other cluster until it is explicitly decommissioned.

After admission, the DSC can only be managed over the network by the Pensando PSMs. The PSM management plane can generate offline authentication tokens to gain access to the DSC and perform restricted operations in cases in which the management network is unavailable.

The Pensando solution may need access to sensitive data, such as private keys for TLS termination, storage encryption keys, etc. The key management system embedded in the Pensando solution is designed to be pluggable and integrate with existing key management systems (KMS) using standard protocols (PKCS#11, KMIP, etc.). An external KMS can also be used to hold keys used to protect sensitive data stored in the centralized PSM configuration, such as user-supplied keys and 3rd party systems credentials.

Authentication and Authorization in the Policy and Services Manager (PSM)

Role-Based Access Control (RBAC) is a core part of Pensando's API and Object model. There are four elements defining the interactions with Pensando's Distributed Services Solution:

- **User**: Users can be locally authenticated or use remote authentication (LDAP, RADIUS)
- **Role**: Role is a collection of permissions. Users can have multiple roles and Administrators can create custom roles for users
- **Permission**: Defines actions that are allowed on a resource type. For instance, a **read** permission for resource type: "Network" will give *read* permission to objects of kind: *Network*
- **RoleBinding**: Role can be assigned to a user through RoleBinding object. For external users, a group concept is used to ease administration of roles to users. Users in LDAP can belong to multiple groups and those groups can be used in RoleBinding to determine roles for external users. Nested LDAP groups are also supported

The RBAC implementation follows a whitelist model. If permission has not been explicitly granted through assigning a role to a user, that user is denied access to the resource. Users, roles and role assignments are scoped within a tenant. A user is not allowed access to resources outside its own tenant.

Authentication Methods

Local: Local users can be created by the PSM administrator. Passwords are hashed using bcrypt and encrypted before being written to disk. Password strength is mandated with minimum length of 9 and 1 digit, 1 special char, and 1 uppercase letter in local user passwords. Customers are encouraged to use ActiveDirectory with the PSM and keep usage of local users to a minimum.

LDAP: Pensando's PSM supports authentication against ActiveDirectory. Secure connections to the LDAP server are achieved through `StartTLS` issued by the PSM.

RADIUS: The PSM supports username/password-based authentication against a RADIUS server.

For customers with existing network and policy management systems that will integrate to the northbound REST interfaces of the PSM, remote machine-to-machine logins are performed, using assigned user roles and permissions.

Authenticated Actions

Upon successful login, the PSM issues a JWT (JSON Web Token), which is signed using HS512 (HMAC using SHA-512) and contains user and tenant info in the claims fields. The validity duration of the token is controlled through a configuration parameter. The Secret used to sign the JWT is stored encrypted on disk.

For each PSM API call, this JWT is validated and user authorization checked before processing the request. For the Policy and Services Manager UI, this token is returned in a secure cookie which is marked as `httpOnly` to protect against Cross-Site Scripting (XSS) attacks. A Cross-Origin check is also performed for POST requests coming from a browser.

Audit Logs

The PSM maintains read-only audit logs for each API call whether performed through UI or CLI. The system records:

- **What** action was done on a resource and what was the outcome
- **When** did it happen (to create a chronological set of records)
- **Who** initiated the action
- From **Where** was it initiated (IP addresses of client)
- On **What** resource was the action done

These logs are maintained for a minimum of 6 months (default) and can be exported for archiving.

About Pensando

Founded in 2017, Pensando Systems is the company pioneering distributed computing designed for the New Edge, powering software-defined cloud, compute, networking, storage and security services to transform existing architectures into the secure, ultra-fast environments demanded by next generation applications. The Pensando platform, a first of its kind, was developed in collaboration with the world's largest cloud, enterprise, storage, and telecommunications leaders and is supported by partnerships with Hewlett Packard Enterprise, NetApp, Oracle, IBM, Equinix, and multiple Fortune 500 customers. Pensando is led by Silicon Valley's legendary "MPLS" team – Mario Mazzola, Prem Jain, Luca Cafiero, Soni Jiandani and Randy Pond – who have an unmatched track record of disruptive innovation having already built $8Bn/Year businesses across storage, switching, routing, wireless, voice/video/data, & software-defined networking. The company is backed by investors that include Lightspeed Venture Partners, Goldman Sachs and JC2 Ventures.

For more information, please visit www.pensando.io