Views on banking-grade digital asset custody solutions

Why it is much more than a "signing machine"

Taurus Group SA

October 2020

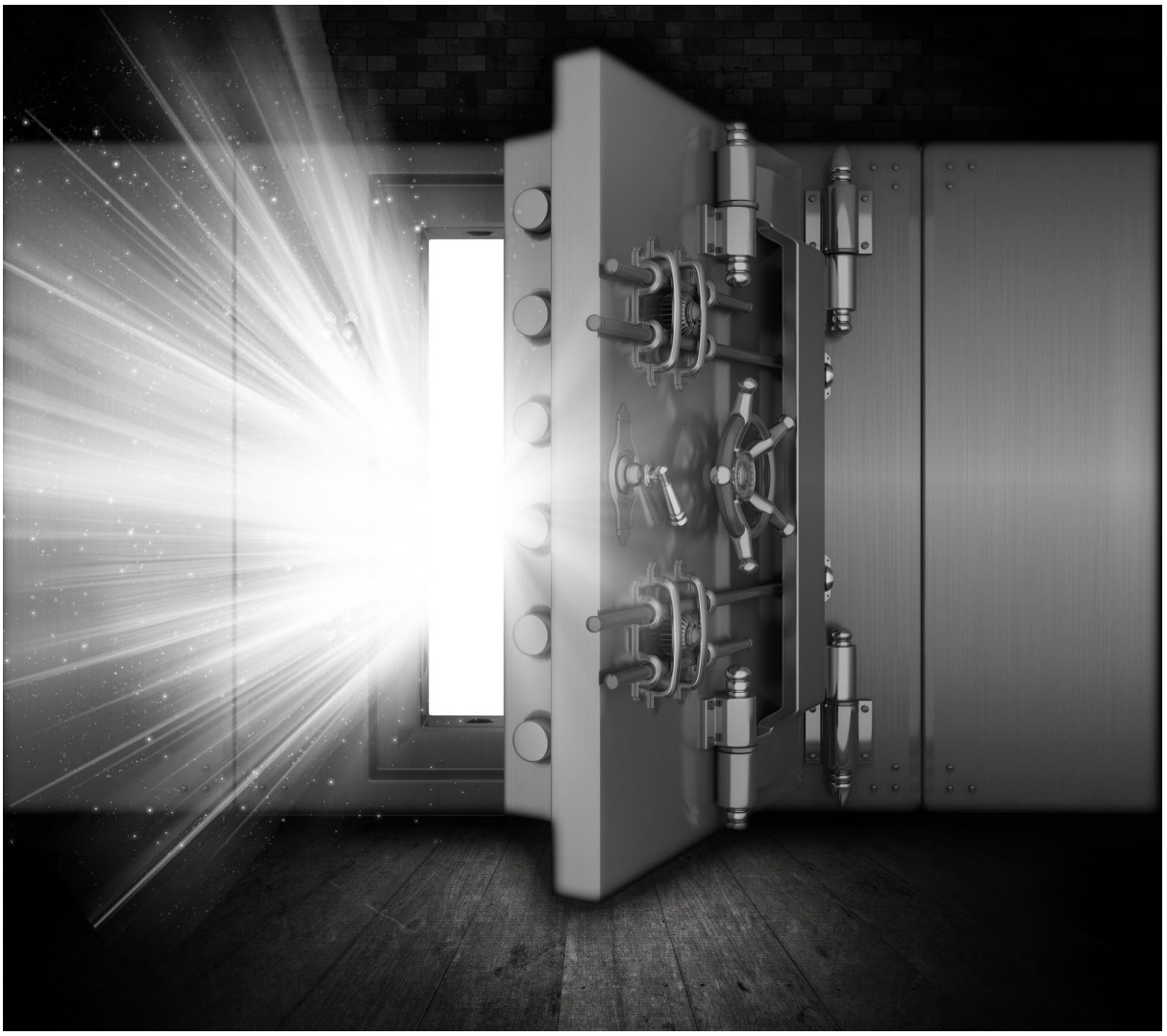


Table of contents

I. Foreword

At Taurus, we have accompanied numerous banks and financial institutions in integrating and deploying digital assets in their business, by providing them with our digital asset infrastructure and technology as well as supporting its integration in banking *production* environments.

We use the same products as our customers for our own activities, be it PROTECT (hot/cold custody), EXPLORER (blockchain nodes), or CAPITAL (issuance, booking, and management of tokenized assets). We thus strive to keep these products superior to alternative solutions with respect to all applicable metrics, be it security, ease-of-use, banking-readiness, and total cost of ownership.

Our engineers have helped leading cryptocurrency exchanges securing their cold wallets, among other projects related to digital assets custody and several management team members have held executive roles in leading financial institutions. We have notably leveraged this experience when leading the creation of the CMTA Digital Assets Custody Standard[1], and when reviewing the security of custody solutions, starting with our own products.

Based on this dual technology and financial services perspective, this article shares our views on what constitutes a modern digital asset custody solution, with a focus on security assurance as well as its technical and procedural aspects. It is written with banking executives (business or technical), auditors, and regulators in mind as target readers. In the remainder of the article, Section II highlights the main components of banking-grade digital asset custody platforms; Section III covers the objectives of such platforms; Section IV concludes with a perspective on multi-party computation technologies.

A one-sentence summary would be the following: production-ready, banking-grade custody platforms should go way beyond simplistic signing machines that new entrants may think of.

Have a good read.

Dr. Jean-Philippe Aumasson
CSO, Co-founder at Taurus Group SA

[1] See www.cmta.ch and https://www.cmta.ch/content/319/cmta-digital-assets-custody-standard-october-2020.pdf.

II. Core elements of banking-grade digital asset custody solutions

Beyond signing machines

A banking-grade digital asset custody solution must address all the needs of a financial institution regarding digital assets storage and management, and is thus much more than a wallet or than a "signing machine". Indeed, besides secret keys' confidentiality, a custody solution must fulfill requirements related to authentication, integrity, availability, accountability, scalability, and business continuity. It must also address a number of functional, operational, business, and compliance needs. To summarize all these needs, we have broken down a custody solution into different layers, as Figure 1 shows, and as briefly described further[2].

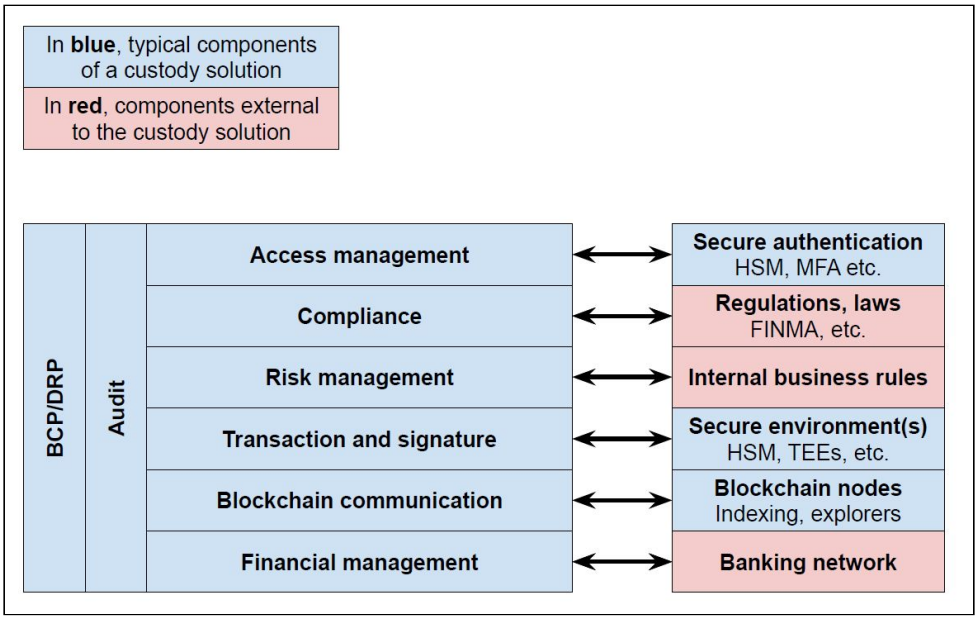


Figure 1: Digital asset custody components & layers .

Access management layer: Reflect any bank's target operating model

- Role-based access control with full audit trail
- API-based and GUI access
- Connectivity with third-party liquidity providers
- Wallet management logic

Compliance layer: Transpose compliance requirements

- KYC and AML checks
- KYT risk scoring and automated fraud detection
- Cryptographically valid proof-of-reserve

[2] This description is not exhaustive, for space and intellectual property reasons.

Risk management layer: Cover operational risk management requirements

- Address whitelisting and blacklisting
- Transaction rules (amount, time of day, etc.)
- Rate-limiting functions (amounts, transaction), configurable at multiple levels
- Emergency kill switch

Transaction and signature layer: Securely build and sign transactions

- Secure seed and key storage
- Blockchain transaction creation
- Transaction signature incl. hierarchical key derivation

Blockchain communication layer: Include a dependable gateway to blockchains

- Often overlooked by providers, who may rely on public nodes with little or no SLA.
- Reliable broadcasting of transactions
- Real-time monitoring of wallet addresses
- Privacy and SLA constraints

Financial management layer: Integrate with bank's business processes and infra.

- Issuance, booking, transfer, and processing of corporate actions of any type of tokenized assets and securities (goes beyond simple cryptocurrency)
- Treasury management incl. nostro and vostro management
- Reconciliation and reporting
- Core banking interfacing and reconciliation

Furthermore, the following layers cover all of the above layers:

Audit layer: Ensure accountability and auditability of the solution

- Audit trail generation and collection
- Audit logs integrity guarantees
- Connection to log management systems

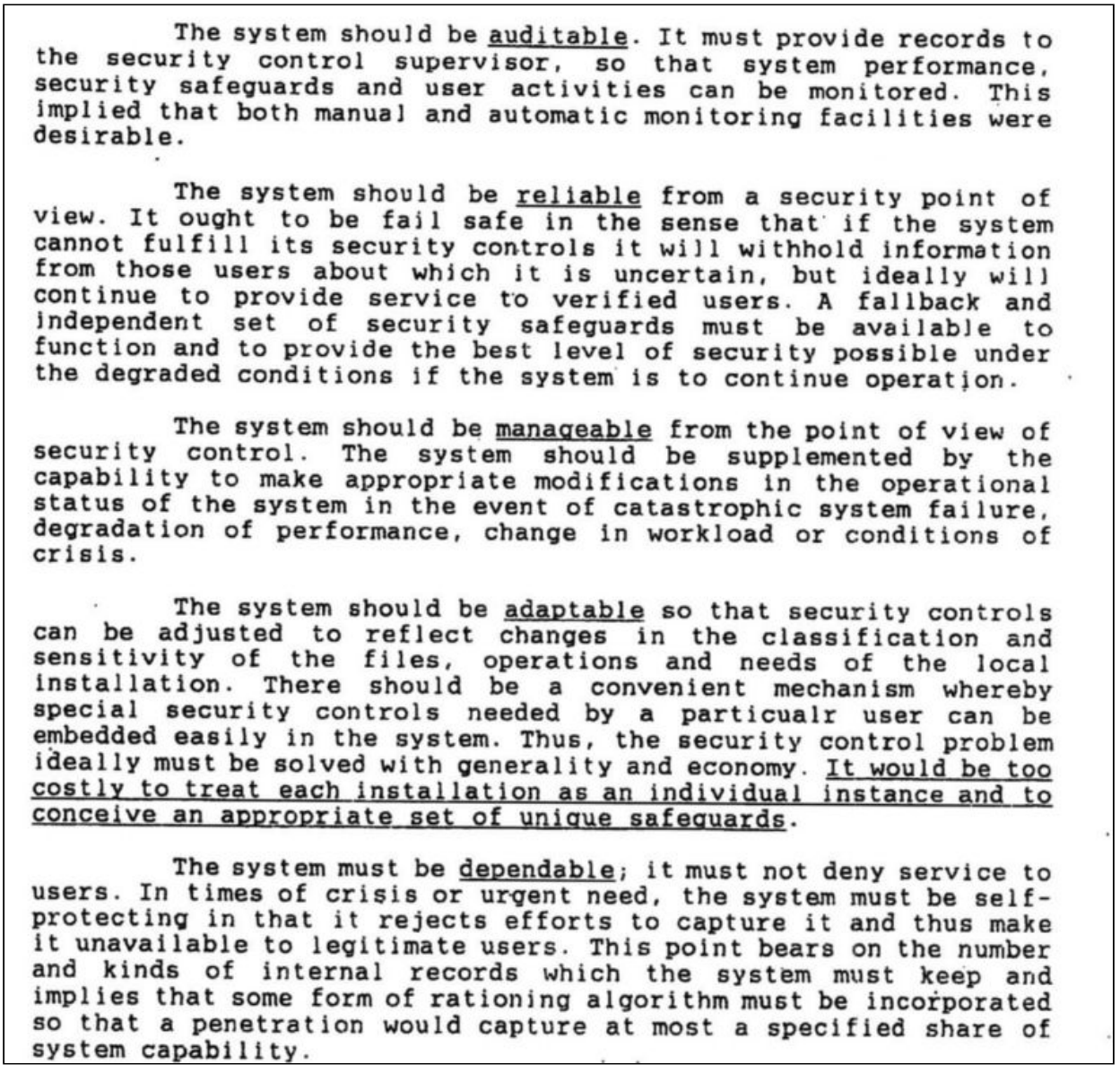
BCP/DRP layer: Permit smooth business continuity and disaster recovery

- Redundant back-ups
- High-availability network services
- Prevention of vendor lock-in

Key take-away: As inferred from the above taxonomy, we strongly believe that a best-practice, *production-ready*, custody solution for banks goes way beyond secure signing and key storage (what we call "signing machines"). New entrants shall not limit themselves to *solving the signature layer* at the risk of having to build suboptimal, unintegrated patches of components from multiple suppliers, or exploding the budget allocated. We will detail in subsequent papers some of the aforementioned layers.

III. The security goals of a custody solution

Besides compliance to banking and regulatory requirements, security is the most important property of banking-grade custody solutions, be it against external attackers or insiders. Providing high security assurance entails multiple and redundant security controls, while mitigating potential components' by following a defense-in-depth approach. Our security architecture notably follows the foundational principles of secure systems, as for example described by NSA in Figure 2.



The system should be auditable. It must provide records to the security control supervisor, so that system performance, security safeguards and user activities can be monitored. This implied that both manual and automatic monitoring facilities were desirable.

The system should be reliable from a security point of view. It ought to be fail safe in the sense that if the system cannot fulfill its security controls it will withhold information from those users about which it is uncertain, but ideally will continue to provide service to verified users. A fallback and independent set of security safeguards must be available to function and to provide the best level of security possible under the degraded conditions if the system is to continue operation.

The system should be manageable from the point of view of security control. The system should be supplemented by the capability to make appropriate modifications in the operational status of the system in the event of catastrophic system failure, degradation of performance, change in workload or conditions of crisis.

The system should be adaptable so that security controls can be adjusted to reflect changes in the classification and sensitivity of the files, operations and needs of the local installation. There should be a convenient mechanism whereby special security controls needed by a particualr user can be embedded easily in the system. Thus, the security control problem ideally must be solved with generality and economy. It would be too costly to treat each installation as an individual instance and to conceive an appropriate set of unique safeguards.

The system must be dependable; it must not deny service to users. In times of crisis or urgent need, the system must be self-protecting in that it rejects efforts to capture it and thus make it unavailable to legitimate users. This point bears on the number and kinds of internal records which the system must keep and implies that some form of rationing algorithm must be incorporated so that a penetration would capture at most a specified share of system capability.

Figure 2: Some of the “general characteristics (..) desirable in a secure system”, as described in NSA’s 1998 *History of Computer Security*[3].

In the following, we discuss some of the specific security goals addressed in our design of digital asset custody systems.

[3] Available at https://cryptome.org/2020/10/nsa-history-computer-security-1998.pdf.

Prevent direct access to the seeds or keys

In a hot/warm setting, seeds or keys may be stored in the secure memory of a tamper resistant system. Taurus-PROTECT for example relies on a hardware security module (HSM) certified FIPS 140-2 Level 3, ensuring that any extraction attempt will be detected by the HSM, which would then erase the secrets. Taurus further enhances the HSM security with its own—proprietary and auditable—firmware extension.

An alternative approach uses cryptographic secret-sharing and multi-party computation (MPC): with such methods, a key is shared among multiple parties, who run a cryptographic protocol to issue a signature without ever exposing the key. This is suitable when no secure hardware is available, but bears a number of limitations (such as the challenge of securing software-only platforms and reliably running share update protocols).

Prevent unauthorized access to signing capabilities

Even if keys are not exposed directly, an attacker could steal funds by having access to the signing module. A custody solution should therefore enforce multi-party approval, typically on a quorum basis[4], in a secure environment such as that of a HSM. Blockchain-specific multi-signature may be suitable for selected use cases, but generally blockchain-agnostic solutions prove more reliable and easier to integrate and scale.

MPC-based methods directly address this problem by cryptographically preventing access to the key for a single party. However, a downside is then that an authorized quorum of parties can directly sign any transaction and recover the key, bypassing potential security controls regarding transaction content and type.

Prevent unauthorized transactions

Even if access to signing capabilities is restricted to authorized parties following a four-eyes principle, said parties may still request transactions unauthorized by the business rules, be it accidentally or maliciously. To mitigate this risk, business processes and rules must be defined and securely applied for the transactions requested. For example, a secure environment such as that of a HSM can enforce security controls regarding transactions' content and approvers' rights.

Generate keys securely

Key generation is much more than picking a safe pseudorandom generator—this is actually the easiest part. Generating keys, or more accurately the seeds that will be used to derive keys, must be done during a well documented and carefully executed key ceremony, during which back-up values are also created, tested, and sealed.

Ensure reliable back-ups

A custody solution with the best cryptography and secure hardware is pointless if someone has access to back-ups of the keys, or if the back-ups will not work when needed. This suggests that back-ups should 1) be subject to shared control, for example via threshold secret-sharing mechanisms, 2) be thoroughly tested during the key ceremony, 3) be stored

[4] The more flexible, the better.

in tamper-evident containers and regularly checked for tampering, and 4) be subject to tested recovery processes and technical procedures, in order to ensure business continuity in the expected time.

Protect logs and databases

Like any modern computerized service, a custody solution generates extensive logs, and performs read and write operations in a database. Database content should be encrypted when relevant (for example if PII[5] data is stored), and the integrity of its content should be protected. Likewise, the integrity of logs is particularly important in order to ensure that any unauthorized operation cannot be erased from the recorded history.

Maximize solution auditability

To ensure a high security assurance, a custody solution should not be a complete black-box. Instead, it should provide a fair amount of transparency regarding its internal logic (for example, by sharing its source code with its users), and regarding its activity (via logs generated by its various components). Control and auditability of the solution is essential in the context of compliance audits, security audits, change management, and incident response processes. The more the provider masters the technology stack, the less opaque/black-box the solution is. Audit reports from reputable third parties should be available to users of the solution.

Ensure supply chain and build integrity

The provider of a custody solution must be able to provide guarantees that the software running in the custody solution, be it on-premise or as a service provider, corresponds to the components provided for audit and cannot be modified during its operation. Such guarantees require a combination of technology (such as continuous integration and authentication tools) and procedures (in terms of change management and role segregation).

Key take-away: Ask your providers what components of the value chain they control, vs. outsourced ones. The highest premium and value shall be given to those who ***master the full technology stack*** including the development and management of cryptographic functions, which represent, in our view, the heart of the nuclear plant. This provides the highest assurance against technology obsolescence – we have seen several providers that assemble this module not being able to cope with technology progress and their clients decide to switch. In other words, identify the builders from the assemblers who outsource the heart of the nuclear plant. At Taurus, we build.

[5] Personally identifiable information.

IV. On the roles of secure hardware and MPC

We are sometimes asked about the merits of multi-party computation (MPC) methods in the context of transaction signature in custody solutions. Some would compare MPC to HSM-based solutions, as if the two approaches were mutually exclusive and if one of the two was inherently superior. We believe that this view is an oversimplification, as the MPC vs. HSM debate only concerns the transaction-signature layer, which is an important layer but just one among others described in section II. As we will discuss, **HSM and MPC achieve different goals, serve different use cases, and may work together.**

The role of an HSM

A hardware security module (HSM) is one approach to create a *trusted execution environment* (TEE)[6], which in the context of a custody solution aims to:

- Securely store cryptographic values (such as blockchain account seeds, keys, but also other values such as authentication tokens, as well as certificates);
- Ensure that the logic executed in the HSM is done as prescribed, and that it does not reveal sensitive information (for example, when applying security controls such as those of the risk management layer, and when creating and signing transactions).

A state-of-the-art HSM notably relies on tamper detection systems in order to reset itself when a threat is detected, as well as code signing mechanisms to only run authorized code. More generally, the use of an HSM facilitates the risk analysis by providing a clearly identifiable trusted computing boundary (TCB), which has been certified to offer a certain level of security assurance. HSMs have been in production for decades for a reason and we believe are here to stay. For example, AWS and Google Cloud Platform provide “cloud HSM” services.

The role of MPC

MPC is a recent development in custody solutions and has a different purpose than HSMs: it is a class of cryptographic techniques that allows a set of participants to collectively issue a signature by running a protocol that does not expose the key. In particular, threshold signature schemes (TSS) allow *t*-of-*n* quorums. MPC therefore focuses on the signature part of the transaction layer. As of today, multiple TSS protocols exist, which differ in terms of adversarial model, security assurance, features (presigning, reshare, abort), protocol rounds (for setup, signing, and other stages).

MPC is thus a tool of choice to mitigate the risk when only software components are used. For example, when a mobile phone application and a server application share control of a key. MPC can thus address selected use cases that purely HSM-based custody may not address, because dedicated hardware cannot be used.

[6] Note that the concepts of TEE and TCB were already rigorously defined in the 1985 “Orange Book”: https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/dod85.pdf.

In principle, the shared control functionality that MPC offers can be achieved with a trusted execution environment, with the additional benefit that security controls can be enforced in a secure environment.

Among current limitations of MPC-based technologies, note that most TSS protocols can only distribute the signing with a given key, as opposed to a full wallet using BIP32 derivation. Furthermore, back-ups as well as a key ceremony remain necessary. MPC and TSS can be nonetheless relevant tools that could even be combined with HSMs to achieve different types of deployment trade-offs.

When assessing an MPC-based solution, one should evaluate whether it matches their scalability needs (2-party vs. multi-party), as well as the operational risk and cost of segregating the different nodes of the system. For example, different nodes, running on different data-centers, must be managed by different teams, such that no single administrator or access control solution can control multiple nodes at the same time (see Figure 3).

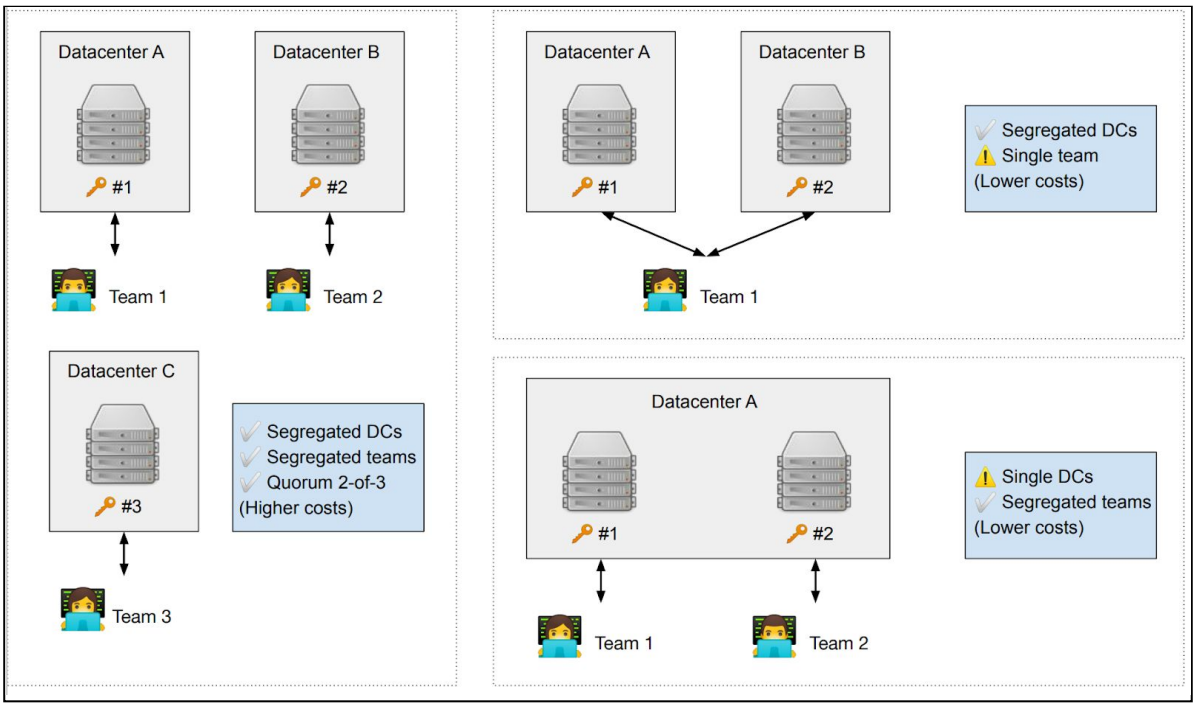


Figure 3: IT components technical and operational segregation models: the most secure and reliable model is the 3-datacenter model (left-hand), which we recommend.

Taurus and MPC

At Taurus, we have implemented HSM-based solutions that we strengthened with our own firmware extension – now in production with the full spectrum of financial institutions[7]. They work and scale well. We have also closely followed the evolution of MPC and TSS

[7] Systemic banks, investment banks, private banks, retail banks, crypto-banks and financial market infrastructure providers.

technologies, assessing the maturity and relevance of known protocols and implementations. In this context, we have notably published some research, such as the article *Attacking Threshold Wallet*[8], and are currently working on concrete projects that you will soon hear about.

That said, one of our conclusions is that a trusted execution environment (such as an HSM) is mandatory for high-assurance digital asset custody, with or without MPC.

Further reading

An effort to establish requirements and recommendations for secure and dependable digital asset custody, aimed for suppliers, customers, and auditors:

- CMTA – *Digital Asset Custody Standard*
  https://www.cmta.ch/content/319/cmta-digital-assets-custody-standard-october-2020.pdf

Historical references on the foundations of security architecture and trusted computing:

- US National Security Agency – *History of Computer Security*
  https://cryptome.org/2020/10/nsa-history-computer-security-1998.pdf
- US Department of Defense – *Trusted Computer System Evaluation Criteria ("Orange Book")*
  https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/dod85.pdf
- Ken Thompson – *Reflections on Trusting Trust*
  https://www.cs.cmu.edu/~rdriley/487/papers/Thompson_1984_ReflectionsonTrustingTrust.pdf

General reference about cryptography:

- Jean-Philippe Aumasson – *Serious Cryptography*
  https://nostarch.com/seriouscrypto

Example of state-of-the-art MPC protocol:

- Rosario Gennaro, Steven Goldfeder – *One Round Threshold ECDSA with Identifiable Abort*
  https://eprint.iacr.org/2020/540

[8] See https://eprint.iacr.org/2020/1052, joint work with Omer Shlomovits from ZenGo (https://zengo.com).