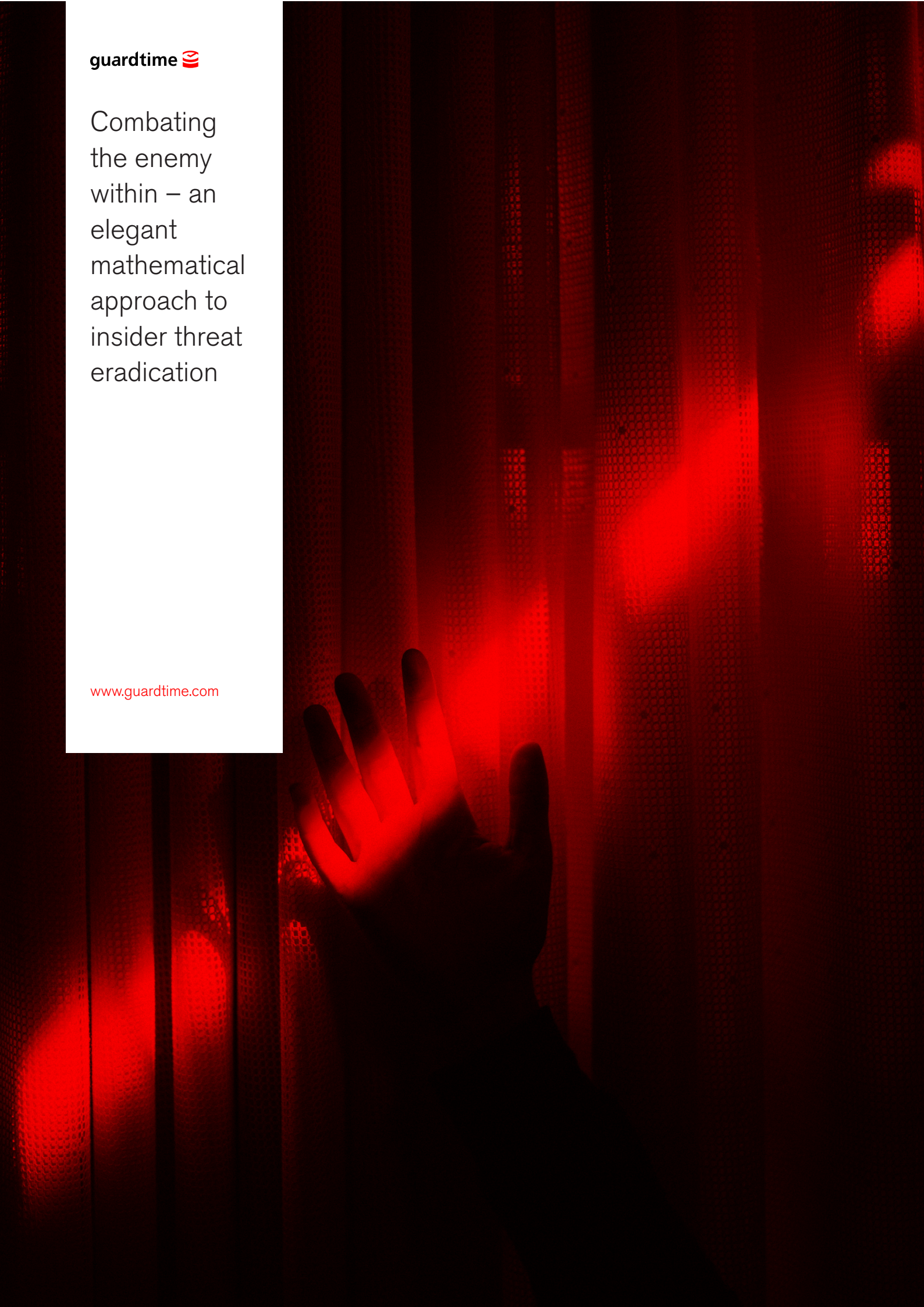
guardtime

Combating the enemy within – an elegant mathematical approach to insider threat eradication



www.guardtime.com

Abstract

"In God we trust, all others we virus scan."
-Author Unknown

The occurrence of insider threats in organizations is not a matter of 'if'; it is a matter of 'when'. Insider attacks on organizations are continuously occurring and in most cases involve simple exploitation of inadequate practices, policies and procedures within the enterprise. The implications for government, financial services, telecommunications and IT industry are profound. If we factor in the continued erosion of traditional layered security boundaries through the increased use of mobile devices (BYOD) and portable storage devices it's not hard to see that mitigating insider threat is getting significantly harder.

There is no magic bullet to prevent or detect insider threats in an organization. This article discusses how Keyless Signature Infrastructure (KSI) enables security professionals to mathematically prove the state of a network or computer asset using hash tree-based real-time authentication schemes, and details some use-cases where such a proof enables detection of malicious activity by privileged users.

By integrating KSI into networks, irrespective of where an asset is transmitted or stored, every component, configuration, and digital asset generated by humans or machines can be tagged, tracked and located with real-time verification independent of trusted administrators. KSI provides a truth-based system wherein the need for trust can be completely eliminated.

Introduction to Insider Threats

Today, organizations where privileged users have access to sensitive information employ traditional role based access control or other established security policies to restrict user actions to reduce the likelihood of sensitive information being compromised. Current network monitoring tools are configured to look for explicit violations of security policies, but data loss nevertheless occurs when 'insiders' intentionally violate policies before detection or are able to remove evidence of their activities from logs and other monitoring systems. Only reliable detection and subsequent intervention can prevent them from wreaking irreparable damage (e.g. the Edward Snowden and Bradley Manning case). Such violations occur in many other situations, including for example financial institutions (e.g. Jerome Kerviel of SocGen). We also know that system vulnerabilities can exist long after organizations are made aware of them (Verizon's 2015 DBIR reports that 99% of exploited vulnerabilities were still compromised more than a year after the CVE was published).

Every year, data loss from insider breaches costs enterprises millions of dollars. Verizon's 2015 DBIR shows that insider threat is the third largest category of incidents at 20.6% with 55% of the incidents being related to privilege abuse.

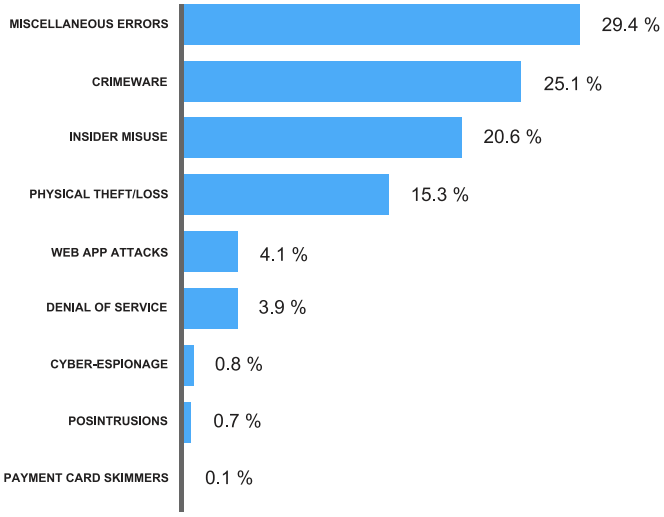


Figure 1 Incident classification patterns - Reference 2015 DBIR

Threat actors fall into three major categories:

- Unintentional and careless insider who accidentally modifies/leaks data.
- Malicious insider who engages in activity with intent of fraud, personal financial gain or grudge.
- Exploited insider who falls victim to manipulation and/or trickery by another malicious entity inside the organization – typically through social engineering.
- External/partner actors

Irrespective of the category, there are often significant delays between the occurrence of malicious activity and its detection using current technologies. (Reference #7)

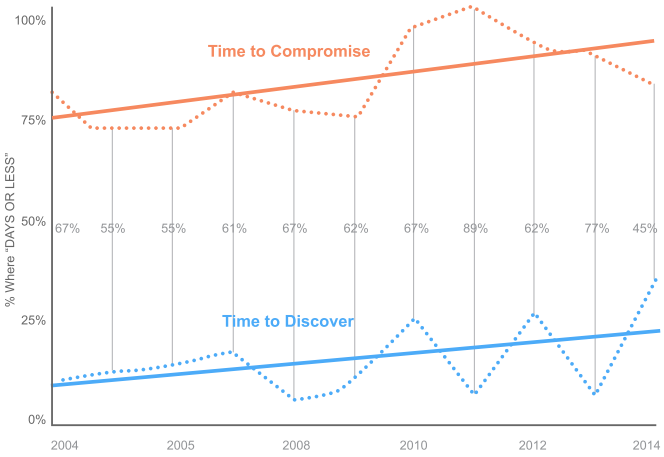


Figure 2 The defender-detection deficit - Reference 2015 DBIR

Insider threat - the problem space

Since the widely publicized disclosures by Edward Snowden, the issue of insider threat has been at the forefront for governments, corporations and security practitioners. Organizations typically take a holistic approach to the insider threat problem, with granular access controls, continuous monitoring and data confidentiality using encryption. However, these solutions can be expensive to deploy and maintain, and do not address the fundamental problem of integrity of the monitoring tools. If one can't trust the threat assessment provided to a Security Operations Center (SOC) or a Network Operations Center (NOC), the protection promised by these security solutions is fundamentally flawed.

Current insider threat detection/prevention tools and technologies typically use statistical driven assessments or a series of hypothesized Boolean (if-then constructs) rules to identify behavioral patterns, and to measure the likley detection time of malicious insider actions on IT systems. However, insider behavior is overwhelmingly dominated by noise that obscures detection (i.e. this 'noise' normally includes many actions that are not indicative of compliant or non- compliant behavior). It rapidly becomes too time consuming or even impossible to filter out irrelevant data. Even when such rules would be effective, an insider with sufficient privileges can often remove the evidence before it is analyzed.

Although identifying and correlating potential indicators of insider threat activity is fundamentally hard, Socio-technical approaches can provide a holistic analysis of various threat indicators. Despite these approaches, findings from insider threat studies across critical infrastructures show that the majority of the attacks are only detected after there was a noticeable irregularity in the system, or after the system became unavailable.

An effective and complimentary approach to deal would be to make the evidence of inappropriate usage immutable – regardless of privilege. To do this, however, there is a need for a technology that can provide evidence of unauthorized changes that cannot be modified or changed without detection. Keyless Signature Infrastructure (KSI) provides such immutable information. KSI can be used to detect the changed state of any digital asset, which can then be investigated or audited. When coupled with correlation and reporting tools this provides real-time awareness, data loss prevention and network attribution, without the need to trust privileged users.

Keyless Signature Infrastructure

What is KSI?

Keyless Signatures Infrastructure (KSI) is a data-centric security technology based on cryptographic hash functions and requires knowledge of only hash-values and binary trees. Every second, a federated and distributed binary tree is generated using hash-values of data generated around the globe within that second. A hash tree is essentially a binary tree of hash values. Two input values, along with any other desired parameters, are concatenated and run through a hash function. This process is iterated, resulting in a single root hash value (Reference #6).

The word "keyless" means that signatures can be verified without assuming continued secrecy of any keys. While shared secrets may still be used for authenticating clients during the signature creation process, no keys are needed for the signature verification itself. The integrity of the signatures is protected using one-way, collision-free hash functions. The collision probability of a typical cryptographic hash function is very small, i.e. it is very hard to find distinct inputs $X \neq X'$ with $h(X') = h(X)$. This is normally referred to as 'collision resistance'.

In the use of KSI, the root hash is calculated and "published" in a distributed "calendar" database that every customer (or subscriber) has a copy of. For every hash value entered into the tree, there is a unique hash-chain, or series of hash-values that allows the root hash-value to be recreated. This hash chain is returned and stored as the signature. A signature for a given digital asset identifies the computation path, through the hash tree, from the asset's own hash value, up to the root calendar value. The signature also includes "sibling" values that were concatenated at every step in the hash tree, which are necessary to recreate the root hash. With access to the public "calendar" database, anyone, anywhere, can receive data and verify the signature, which includes indications of time, identity and integrity, without reliance on a central trust authority.

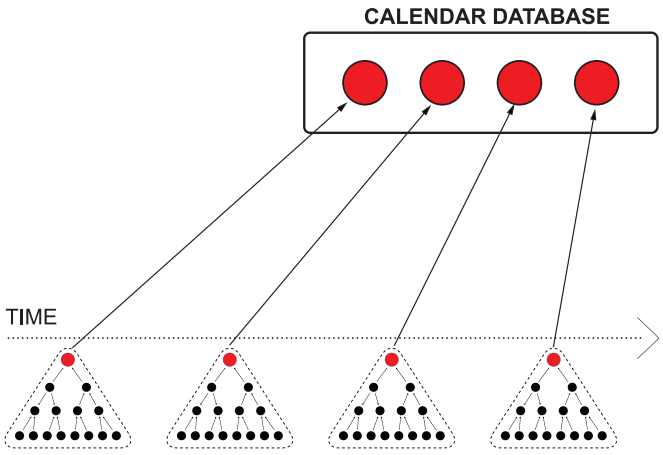


Figure 2: Calendar hash block chain

How does KSI work?

The KSI infrastructure comprises four main components - Cores, Aggregators, Verifiers and Gateways as shown in Figure 3. The core cluster manages the calendar and selects the top root hash for each second. The aggregation network aggregates the hash values and distributes the signatures. The verification network provides widely witnessed access to the state of the calendar. KSI signatures provide proof of signing entities, since parent aggregators accept requests only from authenticated child aggregators.

The hierarchy of aggregation servers creates the global hash tree for each round. Each aggregation server processes requests from the servers below it, adds them to a hash tree and sends the local root hash to the next higher-level server. The servers at each layer wait for responses from the higher-level servers. The first layer of aggregation servers are the gateways that are responsible for collecting and processing requests from clients and then sending the aggregate request to the upstream cluster. This gateway is the customer-facing component of the infrastructure and delivers KSI service to the clients that provide KSI signing and verification services (Reference #2).

Requests are aggregated through multiple layers of aggregator servers and the core cluster chooses the top

root hash. The core operates a distributed state machine which sits at the top of the aggregation network and is responsible for agreeing upon the top root hash for each aggregation period, which it then stores in the calendar database, returning the result to the aggregation network. The regularly spaced rounds used in the aggregation and core processes produce an accurate measure of time, which is embedded into the KSI signature.

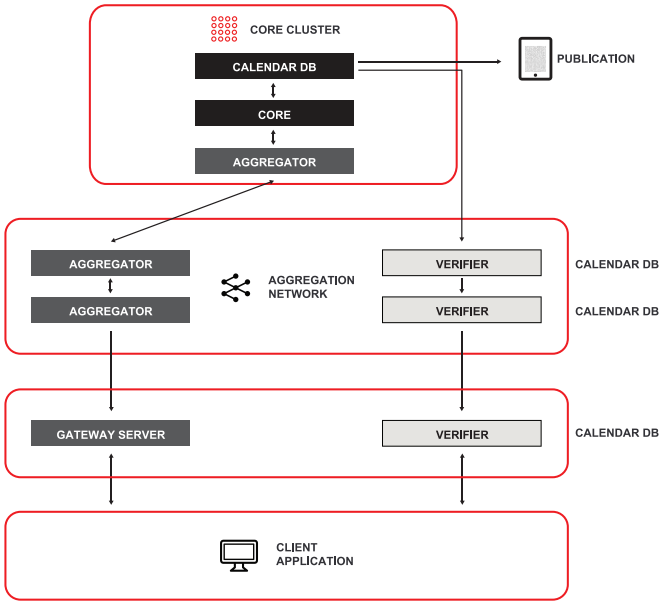


Figure 3: KSI Infrastructure

KSI – System Level Considerations

In contrast to PKI, a KSI based solution is highly scalable, since an increase in the number of signature requests leads only to a distributed increase in the number of hash computations during aggregation. Each aggregator only forwards a single hash value each round, which means that the upstream traffic remains constant, and the computational burden on the core remains the same. To expand the network, one simply needs to add additional servers at the lower levels, mostly as gateways, which are normally client facing. These can be added (as needed) to without affecting other servers or the core. By distributing the calendar downward to aggregators, the burden of actually verifying signatures is also distributed.

On an average, it takes one second for the clients to receive signature responses from the KSI infrastructure, and much less than a second for verification.

How is KSI delivered to an end user application?

Software Development Kits (SDKs) are required to integrate KSI into End-User Applications. Clients who wish to digitally sign objects using KSI use the client side KSI SDKs to communicate with a KSI gateway. The application presents the data hash to the gateway, receives and must then store the signature, and performs verification calls.

How is KSI used to sign digital assets?

KSI can be used to protect any type of digital asset, such as file system objects, virtual machine images, system configuration files, access control files, log files, application and firmware images, and many others.

KSI uses hash tree aggregation techniques. Every piece of data in the ecosystem can be attributed back to a source, whether human or machine. Figure 4 below shows a hash tree computation. The owner (user) sends a hash of a document (or other digital assets such as a log file) to be signed – say x1. All received requests are aggregated into a hash tree as shown below. Signatures comprise of data for reconstructing a path from a leaf node to the top of the tree. For verification the owner of x3 needs x4, x12 and x58 to regenerate the root hash value to prove x3 partic pated in the original computation (see Figure 4).

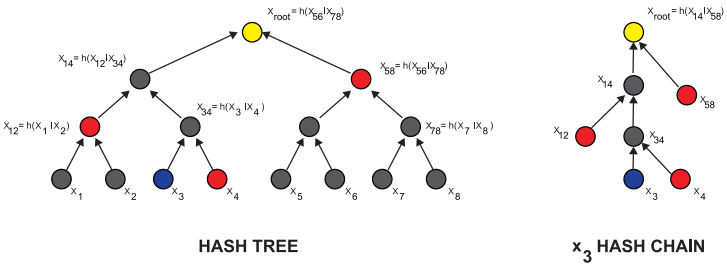


Figure 4: Hash tree aggregation example

How does the calendar blockchain help?

The core cluster maintains the hash calendar. A new tree (with new leaves) is built every second, and each leaf is returned the hash chain to allow it to recreate the public hash value. If a leaf node can recreate the root then the time, integrity and authenticity of the original data can be proven.

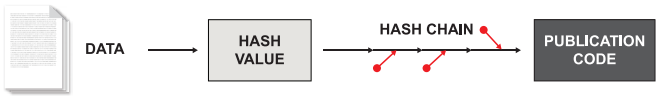


Figure 5: Hash chain signature publication

Only the root hashes are kept in the public calendar database. The calendar block chain is a perpetual hash tree with data only being appended to it (importantly existing nodes are never removed or updated, in the same way an accounting ledger is maintained). The block chain has one leaf for each second since 1970-01-01 00:00:00 UTC. The signing time is encoded in the form of the calendar hash chain. The concatenation order bits encode the path from the root to the leaf and prove the time offset of the leaf from the publication time of the root hash value if the hash function is pre-image resistant (i.e. for all pre-specified outputs it is computationally infeasible to find an input which hashes to that output). No trusted time source is needed.

How can insider threat be mitigated?

To mitigate the insider threat, an understanding of human behavior and motivation as well as a review of critical assets needs to be taken in conjunction with technical solutions. This includes:

KNOWING THE PEOPLE

- Who would target your organization?
- Who are the high-risk individuals in your organization?
- Who has privileged access within your organizations?
- Which business partners pose a risk, based on their access privileges?

KNOWING THE DATA

- What are the critical/sensitive assets in your system?
- Which assets are likely to be the targets?
- Are your systems logs (which track inbound/
- outbound data) integrity protected?
- Do the systems holding those assets have vulnerabilities?

KSI offers a highly scalable data-centric security measure to address the insider threat problem by protecting the critical data assets in the system, both data at rest and potentially in transit. It manages this using widely witnessed evidence - via an industrial-strength block chain.

How does KSI help solve the insider threat problem?

KSI provides a data-centric approach to deter, detect and disrupt insider threat activities. If all critical assets are signed with KSI then - in conjunction with policy enforcement, and appropriate log instrumentation, monitoring and auditing tools, abnormal behavior can be flagged immediately to detect a potential breach, while attempts to remove evidence of such activities can be made impossible. For example, unauthorized file copying can be quickly detected using appropriate policy decisions on the kind of events to log.

Data exfiltration can happen in so many different ways. Insiders could print hard copy of the sensitive data and hand-carry it outside the building, or data could be copied to a thumb drive. The underlying premise behind insider threat detection tools is the interrogation of log file state (where applicable and available) to identify suspicious or anomalous data transfers, and proving that the security controls in place cannot be subverted. Many data exfiltration techniques exist which use data encoding and manipulation to steal data, but if one can guarantee that the log files, access control files, and other critical system configuration files, applications (firewalls, AV systems etc.) cannot be tampered with, then the evidence of these activities can be fully trusted. An integrity failure (implying tampering) can be detected and appropriate incident response measures taken.

Example 1 – Manipulation of log files:

KSI provides the capability to digitally timestamp and sign any type of system event. Since KSI provides proof of history by aggregating hash values and linking them to time (using rigorous mathematics), an insider can no longer hide his tracks by tampering with the log files.

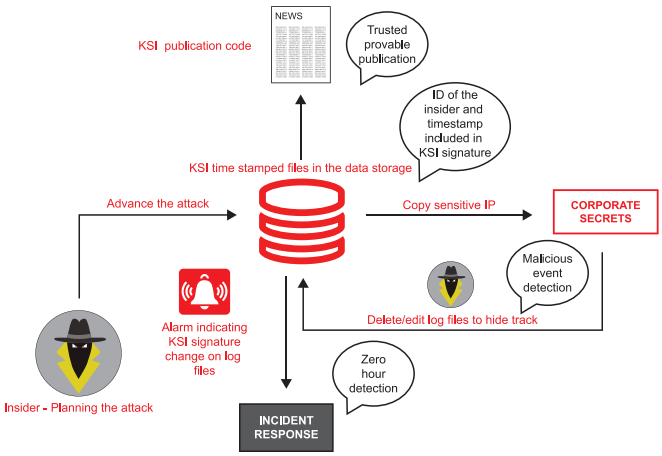


Figure 6: Example of an Insider threat attack via log file manipulation

If the insider copied sensitive company IP and then tried to delete/edit the log files to remove traces of their actions, any software tool that is monitoring the KSI-stamped logs would see a change alert resulting from a failed KSI signature verification. This event can be reported immediately so that the security operations team can take appropriate action quickly. In the absence of technology like KSI, the logs would typically need to be examined manually/visually to interpret changes/malicious events before any action taken, often far too late.

Example 2 – Replacing a legitimate application with malware:

Through a change in signature value, KSI can detect when a malicious entity tries to embed information in otherwise 'clean' files (e.g. through steganography) or tries to replace a legitimate system application with malware. It is also possible to add KSI signatures on sensitive directories, which will enable detection of rootkit or other malicious file insertion.

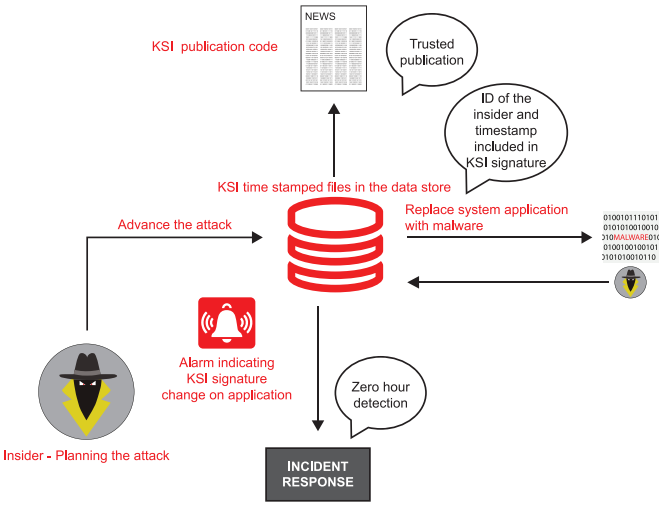


Figure 7: Example of an Insider threat attack using malware

If the insider tried to replace a legitimate system application with a malware-ridden one, any software application monitoring the KSI-stamped applications would see a change alert resulting from the failed KSI signature verification on the application file. This change would be reported immediately so that the IT team can take appropriate action. In the absence of technology like KSI, some of these malicious applications can go undetected, despite use of AV software (for example – The Home Depot and Target breach). It is also feasible to instrument KSI at the OS level to ensure application binaries pass KSI signature verification prior to execution.

Example 3 – Insider collusion:

Even if multiple insiders collude to carry out a malicious operation, KSI can detect this activity, since events cannot be deleted from log files (KSI signature change on the logs causes an alert event). User identity information can be included as part of the signature creation thus providing the ability for an 'innocent' insider to prove they didn't take part in a malicious operation.

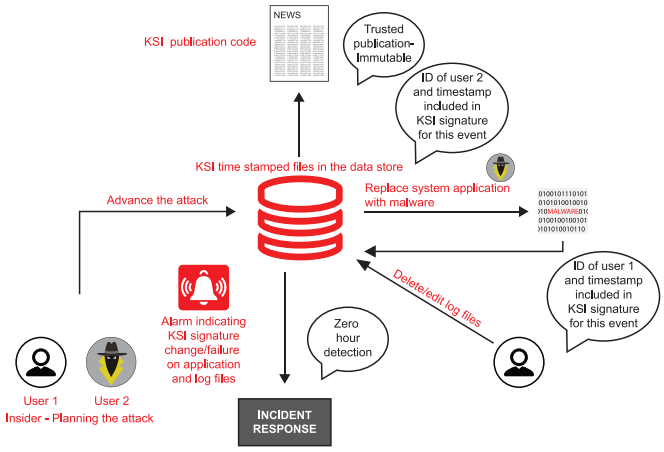


Figure 8: Example of Insider Collusion

If for example several insiders colluded to replace a legitimate system application with a malware-ridden one and then to delete traces of their actions from the system log files, any software tool monitoring the KSI-stamped applications would see a change alert resulting from the failed KSI signature verification on the application file and would be reported immediately. By augmenting KSI with external identity providers (e.g LDAP), the identity of the user performing the action can be included in the signature, innocent employees can prove it was not them who participated in the malicious operation. In the absence of technology like KSI, most current deployments require manual interpretation of system logs/monitoring information to determine the identity of the malicious insider.

Note that KSI is complementary to AV detection; with KSI detection being based on the change in an asset's content, not on the presence of malware. A document master (such as a contract, under strict revision control), might be edited by a user with insufficient authority, either maliciously or in error. The resulting change will trigger an alert from a monitoring tool as explained previously. This

is particularly important in situations where access control is difficult to implement within an organization, and for example in applications such as mergers and acquisitions, or master spreadsheets used in financial services, where strict document control is extremely important.

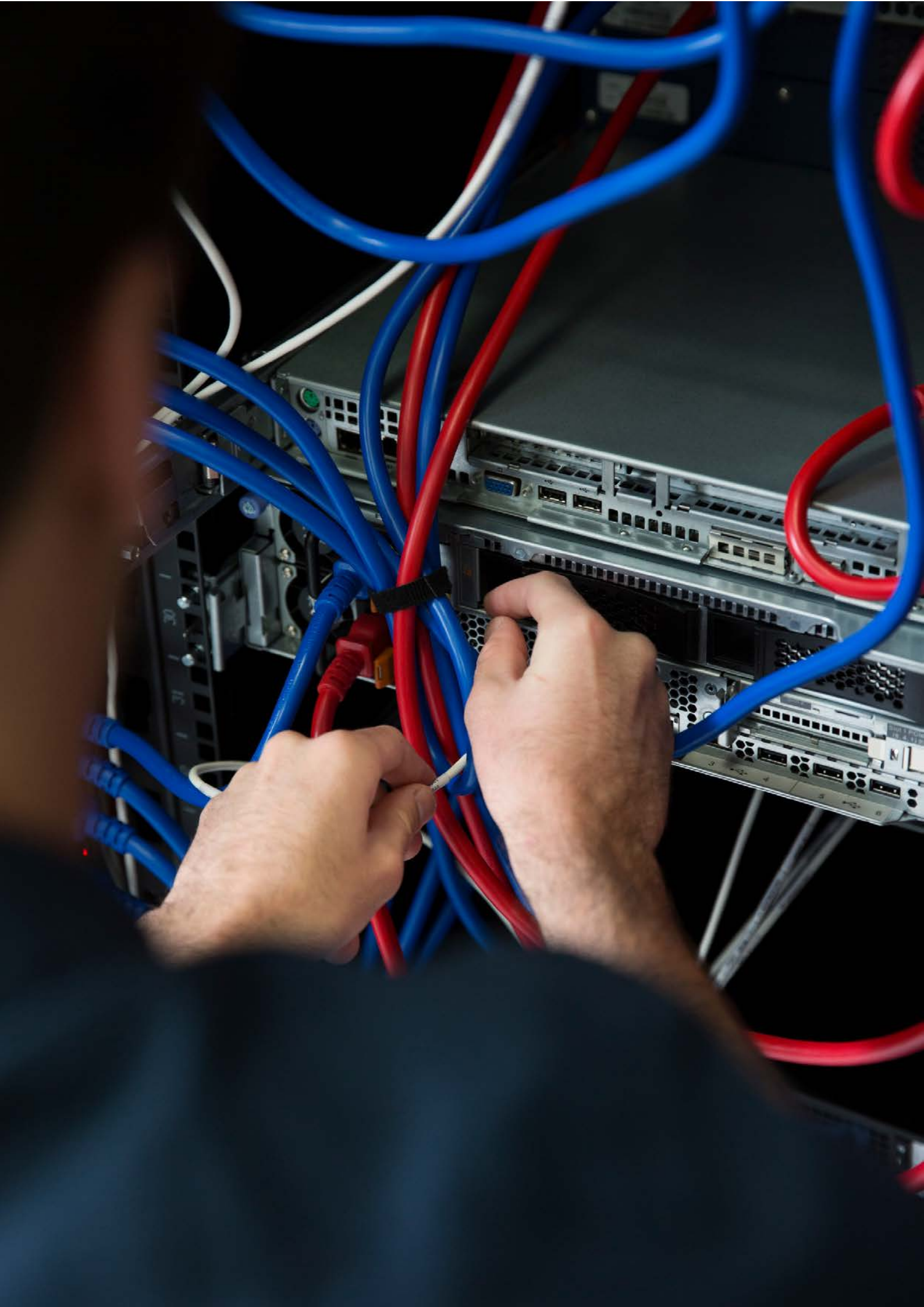
IN SUMMARY, KSI PROVIDES:

- Forensic quality – an immutable chain of custody with independent proof of time, integrity and proof that events occurred in the correct order while ensuring no human interference with the data. This is invaluable to forensic investigators because it provides mathematic certainty of the time and integrity of those logs (Reference #5). It is even possible for external auditors to verify everything that happens to data independently from those who manage the data, since the evidence is completely portable. In the previous examples, the events from planning to execution of the attack will all be time-stamped which provides irrefutable forensic evidence of the sequence of actions.

- Mutual auditability – the enterprise no longer has to trust the service provider, the administrator, and/or auditor to validate the integrity of the evidence. Immutable evidence of authenticity, time, and identity can be preserved for the lifecycle of any digital asset. KSI provides independent proof because it does not rely on humans or any central authority. In the previous examples, all that is needed is the data, hash chain, and the root value, to prove that an insider took part in a malicious action.

- Zero hour problem detection – On average, organizations take 229 days to detect a data breach, according to a recent study from the cybersecurity firm FireEye. (Reference #7)

- Any change to a KSI protected data store is instantly detected through a change in the widely-witnessed keyless signature. Thus, in the above examples, in conjunction with analytic capabilities, KSI provides a new class of information on which an alert can be generated as soon as a malicious activity takes place in the system/network.



How does a KSI-enabled solution differ from other insider threat solutions?

There is no silver bullet for the insider threat problem, but most current insider threat detection controls are woefully inadequate for the job. This is primarily because they are deployed with an external threat actor in mind, and maintaining foolproof access controls to cope with both external and internal users at scale is near impossible. "Trusted" insiders are frequently able to circumvent the access controls and other security mechanisms in place, and remove evidence of their activities. To effectively address the insider threat problem, the solution should be data centric, scalable, and address the "human problem" while providing contextual intelligence and anomaly detection.

Encryption is widely used to provide confidentiality. However, without integrity, encryption brings a false sense of security in cases where malware can be introduced into systems, compromising the integrity of the system securing sensitive data assets. KSI enables auditability and transparency of evidence that in turn offers provable compliance with regulatory and governance frameworks. By focusing on the 'state' of data assets KSI goes beyond AV, and can trigger alerts on any material change to content.

Most existing solutions (including log analysis and SIEM solutions) promote the ability to continuously monitor threat and risk profiles of people in an organization, while maintaining white lists of approved applications. These solutions are not effective unless the system can guarantee, irrefutably, that the logs or applications have not been tampered with, or there is a way to verify beyond doubt that your security measures are working. KSI provides such a verification mechanism for any digital asset. Any change to the asset is detected rapidly via a verification failure of the KSI signature.

Some insider threat solutions rely on profiles of users/applications along with metadata, to compare live access behavior against these profiles. However, when a malicious insider has escalated privileges, the challenge remains to maintain the integrity of these profiles/metadata to ensure such an insider doesn't modify them. Log management techniques are constantly evolving with log collection and sophisticated event correlation being increasingly combined to combat insider threat– indeed log files are a significant data source for activity in the system. KSI can provide integrity on sensitive files and logs. For further protection and assurance of availability, KSI signatures can be protected by an escrow service, with signatures moved offsite to a tamper-proof location if required.

In the case of Target and Home depot, the hackers used stolen credentials to install custom-built malware (that was designed to evade AV) that then stole sensitive customer information. If the system-critical applications had been stamped with KSI signatures, any change to the baseline could have been detected via a KSI signature change alert

KSI HIGHLIGHTS:

- Complements encryption solutions – encryption doesn't mitigate insider threat manipulation of assets – having integrity protection using KSI (so their tracks cannot be covered) does. Knowing that they will be caught may also deter wrongful insider activity.
- Open/offline verification – the KSI calendar information can be downloaded by clients, and hence clients can perform verification based on only publicly available ('widely-witnessed') information - without need for network connectivity.
- No single point of failure - since the core and aggregation network are fully distributed systems.
- Quantum Immunity – organizations worldwide typically rely on PKI for authentication and secure communications. KSI is quantum immune i.e. keyless signatures are resistant to quantum computational attacks, unlike traditional public key cryptosystems like RSA - since they are purely based on cryptographic hash functions that are second pre-image

resistant (Reference #3, #4).

- Mishandling of secrets - unlike traditional public key infrastructure (PKI) signatures, KSI does not require the use of secrets to sign objects or assets. Hence a malicious insider cannot misuse any secrets to hide their tracks.
- The hash chain - contains the information needed to regenerate the root hash value from a given leaf of the tree. The hash chain proves that the input value was part of the original set the tree was built upon. Thus, KSI provides proof of participation of each node in any given hash chain.

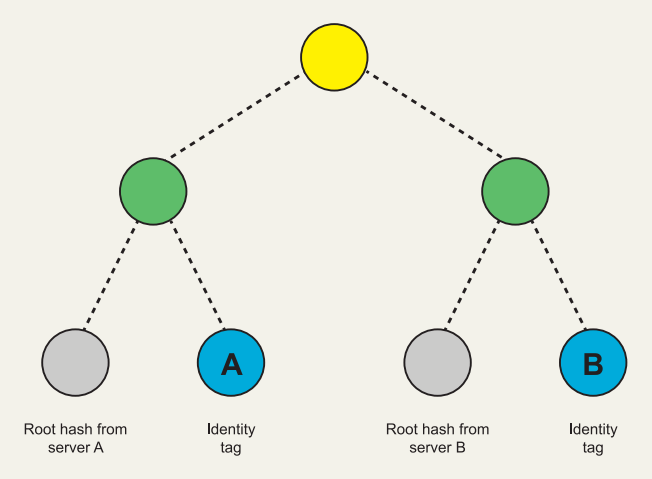


Figure 10: Proving participation using hash chain

- Immutable - the global fingerprint is published electronically every second and also in the world's physical media. Since the publication code (against which the KSI signatures are verified) cannot be tampered with, this ensures that any attempt by rogue administrators to manipulate data can be detected quickly (since it will result in a KSI signature verification failure).

Other KSI Use Cases

- Cloud Security and Forensics - KSI enables real-time authentication against tampering of log files and Virtual Machine images, prior to deployment in the Cloud, thus offering forensic evidence as to the expected state of these assets.
- Storage Integrity – KSI offers independent authentication of data stores
- Executable Integrity - KSI offers a mechanism for protection against tampering of executable files
- Unauthorized Change Control - KSI offers a mechanism for protection against document changes to sensitive documents and files
- Connected Cars - KSI can be used to digitally time stamp application and software files to ensure they are not tampered with. Signing of audit and log files ensures a malicious entity cannot cover their tracks.
- Secure Provenance - KSI offers a means to cryptographically verify ownership of a file/digital object in a way that it cannot be denied by the party modifying the object.
- Enterprise SOC/NOC - KSI instrumented in a security operations center can be used in conjunction with SIEMs and other reporting solutions to monitor critical digital assets.

Conclusion

The application of KSI will materially improve enterprise environments for controlling insider threat and by providing a real deterrent. A potentially malicious insider in a KSI controlled environment will quickly realize that they cannot cover their tracks, and that their activities will be detected and responded to swiftly. This knowledge will also significantly reduce the efficacy of social engineering attempts on peers. Advanced dashboards can be built to extract KSI attributed information from the system and promote custom integration with legacy SIEMs.

KSI-based detection and attribution is widely witnessed owing to the calendar publication and hence it is possible for malicious activity to be provably detected and communicated before the insider even leaves the premises; thus deterring others from attempting similar acts. The ability to rapidly extract chain of custody evidence without exposing critical information from the digital assets under KSI-protection makes same-day forensic responses now possible.

As enterprises further utilize cloud services, KSI helps them stay one step ahead of malicious insiders. Insider threats start with the propensity of a an insider to engage in malicious acts, followed by a planning and execution phase. KSI deters insiders who have a propensity to engage in malicious acts, since it provides digital timestamps that cannot be forged. All critical components in the network are essentially attributable, and the evidence of interactions between users and these assets immutable. With KSI, you can continue to trust your administrators and users, but more importantly you can now independently verify their actions.

100% crime prevention is impossible, however it is now possible to have 100% detection, accountability and auditability, and across highly complex systems. Whilst an effective solution to insider threat has so far proved elusive, KSI now offers a truly scalable solution based on mathematical certainty. Where human motivation and behavior must to be verified in conjunction with effective security controls - think KSI.

References

1. 2015 Data Breach Incident Report - http://www.verizonenterprise.com/DBIR/2015/

2. Ahto Buldas, Andres Kroonmaa, and Risto Laanoja: Keyless Signatures' Infrastructure – How to Build Global Distributed Hash-Trees, Cryptology ePrint Archive –Report 2013/834, https://eprint.iacr.org/2013/834.pdf

3. Ahto Buldas, Risto Laanoja, AhtoTruu: Efficient Quantum-Immune Keyless Signatures with Identity, Cryptology ePrint Archive- Report 2014/321, https://eprint.iacr.org/2014/321

4. Ahto Buldas, Risto Laanoja, AhtoTruu: Efficient Implementation of Keyless Signatures with Hash Sequence Authentication, Cryptology ePrint Archive – Report 2014/689, https://eprint.iacr.org/2014/689

5. Ahto Buldas, AhtoTruu, Risto Laanoja, Rainer Gerhards: Efficient Record-Level Keyless Signatures for Audit Logs, Cryptology ePrint Archive- Report 2014/552, https://eprint.iacr.org/2014/552

6. http://en.wikipedia.org/wiki/Hash_calendar

7. https://www2.fireeye.com/rs/fireye/images/rpt-m-trends-2015.pdf

guardtime