SANS

A SANS Whitepaper

Zero Trust: What You Need to Know to Secure Your Data and Networks

Written by **Dave Shackleford**

March 2020

Sponsored by:

Gigamon

Introduction

Security professionals are rethinking the way we're approaching network and data security. We've realized we need to:

- Look at our entire environment as untrusted or compromised, in addition to thinking in terms of "outside-in" attack vectors. Increasingly, the most damaging attack scenarios occur from advanced malware and phishing resulting in an adversary's malicious access to data from compromised internal hosts and 'trusted' users.
- Better understand application behavior on the network and at the endpoint, and look at what types of network communication approved applications really should be transmitting.
- Reduce or remove implicit trust relationships and system-to-system relationships in general within all parts of our environment. Most of the communications we see in enterprise networks today are either wholly unnecessary or not relevant to the systems or applications needed for business.
- Emphasize data-centric approaches to protection that align with classification types and processing environments.

These are all worthwhile goals, but many of our traditional controls require additional technologies or processes to accomplish them. Compounding this challenge is the advent of highly virtualized and converged workloads, as well as public cloud workloads that are dynamic in nature. Cloud workloads may move between on-premises and external cloud service environments or between various segments within a cloud service provider environment.

The nature of workloads is changing, too. For instance, it's rare for a workload to be uploaded to AWS or Azure and remain untouched or unmoved. The ongoing movement toward increasingly hybrid software-based environments has caused enterprises designing dynamic security architecture models to start adopting an overarching theme: one of "zero trust." As with any evolving architecture or design concept, there is still work to do to define the core elements of a well-rounded zero trust model. Along with the well-known elements of a data plane and a control plane in proposed zero trust architectures, the critical missing element to securing your data and network is the inclusion of a complete monitoring plane, which we explore in this paper.

Zero Trust Defined

What is zero trust, exactly? Zero trust is a model where data, rather than devices and users, is the central focus of all isolation and protection tactics, and all assets in an IT operating environment handling sensitive data are considered untrusted by default until network traffic and behavior is validated and approved. Initially, the concept meant segmenting and securing the network across locations and hosting models. Today though, there's more integration into individual servers and workloads to inspect application components, binaries and the behavior of systems communicating in an application architecture.

Zero trust is a model where data is the central focus of all isolation and protection tactics, and all assets in an IT operating environment that handle important data are considered untrusted by default until network traffic and behavior is validated and approved.

The zero trust approach does not involve eliminating the perimeter; instead, it leverages network micro-segmentation to move the perimeter as close as possible to privileged apps and protected surface areas. It also includes continuous assessment of identity relationships and privileges in use. To date, there has been a wide variety of approaches taken to achieve this overarching concept of "zero trust," but the concept is still evolving and maturing.

Elements of Zero Trust

The recently released NIST draft publication SP800-207,[1] focused on a zero trust architecture model, includes the following elements in a comprehensive data/network security strategy that meets zero trust principles:

- **Identity—**Role and privilege definitions for user/account access
- **Credentials—**Authentication controls such as passwords and keys
- **Access management—**Controls and policies that govern what assets and services can be accessed and from where

[1] "Zero Trust Architecture," National Institute of Standards and Technology, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207-draft2.pdf

- **Operations—**The overarching tools and processes needed to define, implement, maintain and monitor zero trust architectures
- **Endpoints—**Distinct systems and workloads that are part of a zero trust environment
- **Hosting environments—**The environment where a zero trust architecture is implemented (for example, a data center or cloud provider infrastructure)
- **Interconnecting infrastructure—**Tools and platforms facilitating connectivity to and from assets both within a zero trust architecture and externally

With these components and principles as a foundation, zero trust architectures have focused on foundational controls that can facilitate limiting access to systems and data (see Figure 1). Zero trust as a concept is still evolving, however, and there are a number of controls suggested as core architecture elements.

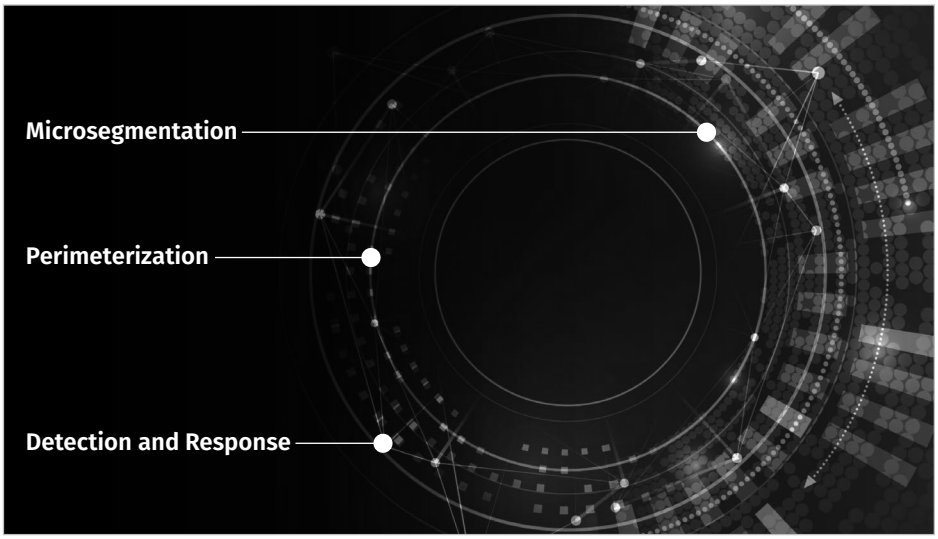

Figure 1. Zero Trust Foundational Elements

Network Access Control: Microsegmentation

The first major component of a traditional zero trust design model is network segmentation closely aligned with a specific type of system or workload (often termed microsegmentation). Zero trust's traditional concept of network microsegmentation strives to prevent attackers from using unapproved network connections to attack systems, move laterally from a compromised application or system or perform any illicit network activity regardless of environment.

By reducing lateral movement, a zero trust microsegmentation model also reduces the post-compromise risk when an attacker has illicitly gained access to an asset within a data center or cloud environment. Security architecture and operations teams (and often DevOps and cloud engineering teams) refer to this as limiting the "blast radius" of an attack, as any damage is contained to the smallest possible surface area while attackers are prevented from leveraging one compromised asset to access another. This reduction in movement is made possible by properly understanding application traffic flows to resources housing important data, understanding the behaviors of the applications and enforcing the proper restrictions. For example, if an application workload (web services such as NGINX or Apache) is legitimately permitted to communicate with a database server, an attacker trying to gain access would have to compromise the system and then perfectly emulate the web services when trying to laterally move to the database server (even issuing traffic directly from the local binaries and services installed). While microsegmentation will reduce risk, credential theft and reuse remain the largest threat to any zero trust architecture.

Identity "Perimeterization": Users and Access Rights

Another foundational element of a zero trust model is that of identity and access management (IAM), specifically focused on users and role-based access to and integration with applications and services. As most application and service interactions have some tie-in to roles and privileges (users, groups and service accounts), ensuring that any zero trust technology can interface with identity stores and policies in real time and enforce policy decisions on allowed and disallowed actions makes sense. IAM is a huge area to cover, encompassing everything from user directories to access controls, authentication and authorization. To facilitate a secure, restricted model, which is at the heart of a zero trust design, IAM should be considered a primary area of focus and resource commitment.

According to the NIST draft, there are certain tenets of a zero trust architecture that must be in place during design and deployment:[2]

1. **All data sources and computing services are considered resources.** While somewhat self-explanatory, this principle forms the basis of policy, defining resources and data sources that will have access control models built around them.
2. **All communication is secured regardless of network location.** This principle primarily focuses on protecting network traffic through the use of encryption and other technology controls.
3. **Access to individual enterprise resources is granted on a per-session basis.** In keeping with the zero trust theme, each attempted connection is vetted and evaluated against defined policy before access is granted.
4. **Access to resources is determined by dynamic policy, including the observable state of client identity, application, and the requesting asset, and may include other behavioral attributes.** As highlighted earlier in the paper, identity is a core aspect of zero trust in every way and forms the basis of many policies and access decisions (along with other behavioral aspects of connections like location, system labels and types and data types).
5. **The enterprise ensures that all owned and associated devices are in the most secure state possible and monitors assets to ensure that they remain in the most secure state possible.** This component of zero trust focuses on system and service configuration and lockdown, as well as some degree of monitoring to ensure the desired configuration state is maintained over time.
6. **All resource authentication and authorization are dynamic and strictly enforced before access is allowed.** In alignment with the previous item focused on IAM, user access will need to have authentication controls in place that are dynamic and integrated into policy decisions.
7. **The enterprise collects as much information as possible about the current state of the network infrastructure and communications and uses it to improve its security posture.** An enterprise would collect data about network traffic and access requests, which is then used to improve policy creation and enforcement. This data can also be used to provide context for access requests from subjects.

[2] "Zero Trust Architecture," National Institute of Standards and Technology, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207-draft2.pdf

Essentially, zero trust facilitates the creation of "affinity policies", where systems have relationships, permitted applications and traffic—any attempted communications are evaluated and compared against these policies to determine whether the actions should be permitted. This happens continuously, and effective zero trust control technology should also include some sort of machine learning capabilities to perform analytics processing of attempted behaviors, adapting dynamically over time to changes in the workloads and application environments.

Emerging: Detection and Response

The original elements of zero trust are focused on the reduction or removal of implicit trust in access to data. For a comprehensive zero trust model, however, another major area of emphasis should be incorporated into the framework: detection and response. Simply focusing on access control is not enough in today's highly dynamic environments—a zero trust architecture needs to incorporate automated detection and alerting of threats with high fidelity, feeding alerts and alarms to a monitoring team such as the Security Operations Center (SOC). Triage can then be performed.

This feedback loop, as well as the automated model of potential response activities that could come from zero trust architecture designs, can help organizations detect and respond to a variety of threats more efficiently and effectively.

The Missing Link: The Monitoring Plane

There are three fundamental elements of a zero trust framework/architecture. First and foremost, to successfully architect a zero trust model, deep continuous network monitoring must be performed. Unfortunately, a dedicated network monitoring plane is not outlined in common zero trust architecture frameworks. It is a foundational element that allows for discovery, deployment, detection and response. The second element is the data plane, which is where organizations' critical data resides and is accessed by resources (systems), applications and users. The third and most commonly discussed element of a zero trust framework is the control plane. The control plane facilitates the reduction (or removal) of implicit trust when accessing data. Each is detailed in Table 1.

Table 1. Elements of a Progressive Zero Trust Framework

Element	Purpose
Monitoring Plane	A monitoring plane provides the foundation for: • Identifying and classifying your data • Mapping flows of your sensitive data • Understanding your network, devices and applications • Performing continuous monitoring for malicious activity, detection of threats and facilitating mitigation actions Key Component Examples: • Network Packet Brokers • Network Traffic Analytics • Application Intelligence and Filtering • Threat Detection Systems • Network Forensic Metadata Collectors • SIEMs

Element	Purpose
Data Plane	The data plane is where your important data resides. It will include resources (systems), applications and users who interact with the data. Here you should build your micro-perimeters around your important data and implement gateways that challenge each interaction with the data for proper, continuous authentication and authorization by policy. Key Component Examples: • Identity Access Management Systems • Strong Authentication Factors • Firewall Gateways / Network Access Controls
Control Plane	The control plane is where your business rules are created and maintained to reduce (remove) implicit trust. The control plane will administer the policies by interacting with the gateways that protect your important data. The control plane should also function to facilitate response policies based on detection of potential threats. Key Component Examples: • Policy Engines and Authorizations • Authentication Stores and ID Management

Stages of Zero Trust Implementations

Despite other zero trust frameworks implying the need for a monitoring plane, it is often an unfortunate omission leading many to the erroneous conclusion that zero trust deployments with microsegmentation and strong identity validation alone are sufficient. To that end, a distinct monitoring plane is an essential part of a mature zero trust architecture that can facilitate *continuous discovery* and monitoring, as well as *detection and response*. A breakdown of key stages (see Figure 2) in any zero trust implementation should include the following:

Figure 2. Key Stages of Zero Trust Implementation

- **Discovery: Discover, catalogue and classify data and assets—**The discovery phase is one of the more important phases in a zero trust architecture model because the different types of assets, traffic flows and data in any environment will need to be continually found and assessed against defined policies. In most traditional data centers, discovery has proved challenging at scale due to a lack of cohesive visibility into all network segments. In a zero trust environment, discovery should focus on network monitoring that discovers, catalogues and classifies data in all storage and application deployments. Mapping data flows for sensitive data scenarios is another important function discovery tools should facilitate.

A distinct monitoring plane is an essential part of a mature zero trust architecture.

- **Deployment: Micro-perimeters and architecture—**In a zero trust deployment, some sort of microsegmentation engine must be in place to enact access control policies defined by a central policy engine. This engine may include cloud-native microsegmentation tools such as Amazon EC2 Security Groups, as well as internal identity-aware policy engines that can restrict and limit access between assets running in on-premises data centers as well as cloud provider environments. Once identities are challenged, confirmed and validated (through integration with directory and other identity stores), a least-privileged authorization access model should be enforced through policy.

- **Detection: Network and application traffic monitoring—**A monitoring plane is needed to continuously detect and track network traffic in the environment, mapping usage of applications, services exposed, user interaction models and patterns of network behavior in expected and unexpected application usage scenarios. In a zero trust model, it is not enough to remove implicit trust in user and system relationships. You must also assume your zero trust approach is not impenetrable and have focused efforts to detect adversaries within your environment.
- **Response: Policies and automation actions—**In today's dynamic environments, automated response actions are becoming more commonplace for specific use cases and playbooks. Response actions can be "triggered" through continuous monitoring and detection of events and behaviors likely indicating compromise or attempted compromise, and may include quarantine of assets, suspension or deletion of systems and workloads, resegmentation of networks and traffic flows, suspension or removal of privileges or user accounts and identities and more. For this to happen at scale, powerful analytics and environment integration are likely needed for most enterprises.

Zero Trust Technology and Evolution

How should organizations go about implementing a robust zero trust architecture including a policy engine, brokering and traffic control and monitoring tools, and detection/response controls and capabilities? The following tools, technologies and tactics are important in the implementation of a zero trust environment incorporating a distinct monitoring plane (see Figure 3).

- **Packet brokers—**Packet brokers are tools acting as network interception and monitoring engines, facilitating traffic capture and/or redirection for analysis by any number of different monitoring and network forensics tools. Modern network packet brokers can direct traffic to a single location or multiple tools simultaneously and should be capable of handling numerous inbound streams in high-speed network environments. Modern packet brokers can also remove redundant data at wire speed if desired and may also include application layer and protocol behavioral inspection and analytics. To implement a zero trust architecture, using packet brokering tools should really be a requirement for

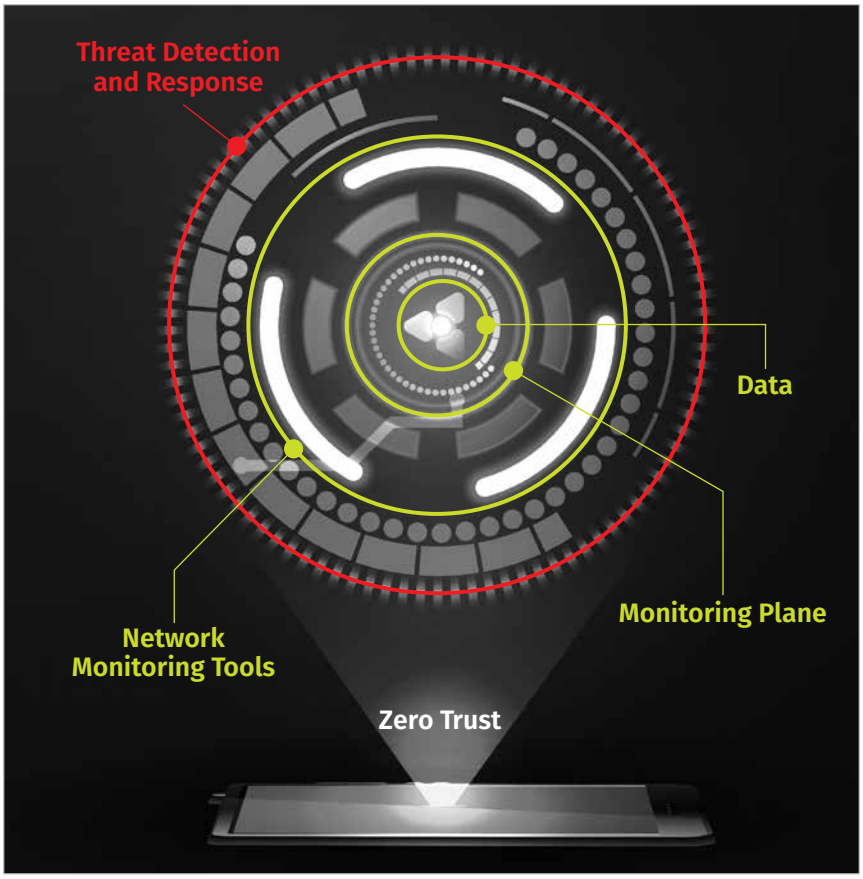

Figure 3. Important Tools, Technologies and Tactics for Successful Zero Trust Implementation

decrypting SSL/TLS traffic at high speeds to perform analysis. Without this type of network visibility, enacting and enforcing policies defined by a policy engine may prove extremely challenging. For privacy reasons, modern traffic brokering tools should be able to mask or filter traffic, too.

- **Network monitoring tools—**Although modern packet brokers often include analytics for monitoring, most enterprise security teams will likely employ a variety of monitoring tools for deep packet analysis, network forensics, traffic "recording" and full packet or network traffic meta-data capture and network intrusion detection. Most zero trust best practices have placed heavy emphasis on the endpoint and identity-based controls discussed earlier, but to present a comprehensive picture of zero trust that accommodates for potential failure in those control areas, real-time network monitoring has to be included.
- **Threat detection/response—**To round out the comprehensive control set in a progressive zero trust architecture, automated or semi-automated detection and response capabilities should also be integrated into the infrastructure. Ideally, these controls leverage all the major elements of a zero trust architecture (endpoint, identity and network), and help organizations detect specific attack scenarios such as:
  - **Lateral movement—**Lateral movement between systems is a common scenario in today's attack campaigns, where initial ingress into a network environment is usually followed by probes and attempts to compromise additional systems nearby. Detection and prevention of malicious lateral movement requires an understanding of your micro-segmentation, authorized identities and normal traffic flows within the environment, as well as real-time monitoring and response capabilities.
  - **Insider threats—**Insider threats are notoriously difficult to detect, as insiders usually have access already granted, thus limiting the efficacy of identity or endpoint-specific controls in a zero trust design. Only deep understanding of expected and unepected behaviors in specific user-oriented interaction with data and applications can help in detecting insider threats.
  - **Credential theft—**Credential theft is a common attack strategy often resulting in lateral movement and user/account impersonation scenarios. Credential theft can be difficult to detect without behavioral monitoring controls.
  - **Data exfiltration—**Data exfiltration usually occurs over encrypted channels such as TLS, making traffic interception and analysis critical in detection and response.
  - **Command and control (C2)—**Similar to data exfiltration, command and control traffic can be challenging to detect and eliminate without real-time visibility into network behavior.

As a zero trust deployment matures and becomes more stable, organizations will want to determine how effective the approach has been in their environments. Industry-wide zero trust metrics are few and far between currently, as most organizations are still in early phases of deployment and have not yet developed mature, long-term strategies for this type of architecture.

Wrapping Up: A Complete Picture of Zero Trust

The NIST SP800-207 draft architecture reference acknowledges the importance of network visibility to a zero trust architecture design but makes no explicit mention of a monitoring plane requirement. Even without full SSL/TLS decryption, network metadata can be collected and analyzed to help detect anomalous patterns and attacks, but a full-featured and comprehensive zero trust architecture should include a control plane with policy definition, enforcement controls and tools, a data plane where systems and data reside (as well as possible identity stores) and a monitoring plane incorporating network brokering and monitoring:

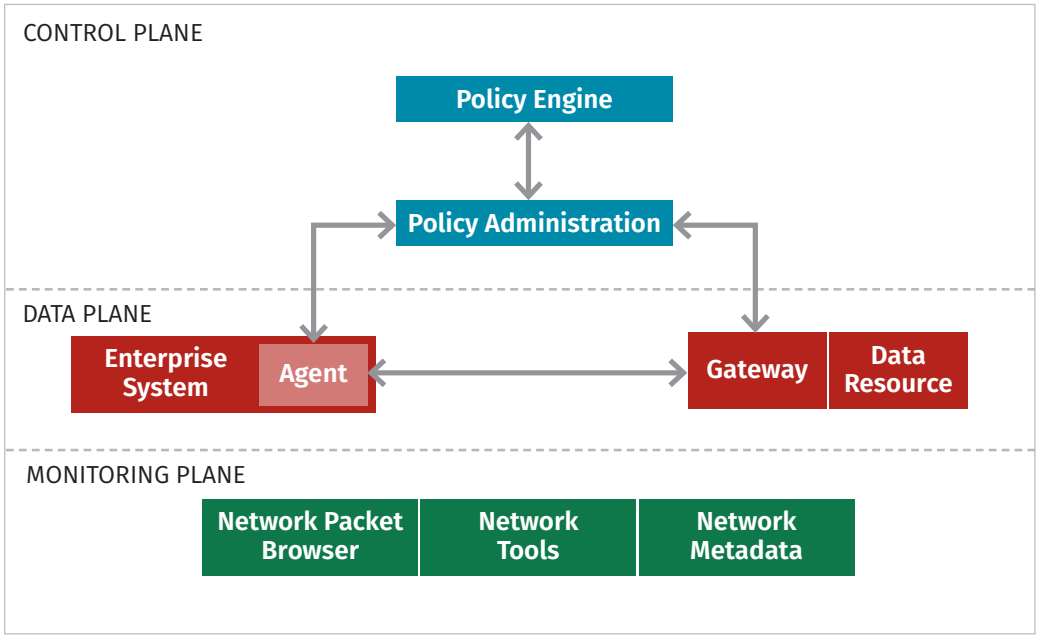

Figure 4. Zero Trust Architecture with Control, Data and Monitoring Planes

Many organizations already have one or more elements of a zero trust model deployed, but these controls are often not working in tandem. Overall, there are a number of general best practices organizations should keep in mind for implementing zero trust tools and controls. Such practices include the following:

- Start with passive application discovery, usually implemented with network traffic monitoring. Allow for several weeks of discovery to find the relationships in place and coordinate with stakeholders who are knowledgeable about what "normal" traffic patterns and intersystem communications look like. Enforcement policies should be enacted later, after confirming the appropriate relationships that should be in place, along with application behavior.
- Design zero trust architecture based on how data moves across the network and how users and apps access sensitive information. This design will assist in determining how the network should be segmented and where protection and access controls should be positioned using virtual mechanisms and/or physical devices between the borders of different network segments.

Examples of Metrics We've Seen Considered and In Use

- Discovery: Number of applications identified and mapped (often tracked per network range, business unit or geographic locale)
- Discovery: Number of tracked and labeled identities, network segments, asset inventory, application inventory and flows of sensitive data
- Percent reduction in network access control alerts (after implementing enforcement policies)
- Percent reduction in compromised systems and application workloads after zero trust implementation
- Percent reduction in incidents and/or mean time to detect or mean time to respond

A full-featured and comprehensive zero trust architecture should include a control plane with policy definition and enforcement controls and tools, a data plane where systems and data reside and a monitoring plane incorporating network brokering and monitoring. One example of this type of architecture is shown in Figure 4.

- More advanced zero trust tools integrate with asset "identities" (which may be part of an application architecture, aligned with a business unit or group or representative of a specific system type). Take the time to categorize systems and applications, which will help in building application traffic baselines and behaviors.
- Ensure you have detection and response solutions that study network traffic, identify malicious signals, allow for investigation of activity and facilitate automated response actions.
- Look for products that work in both internal and public cloud environments where possible—this will almost always require an agent-based solution.

A zero trust architecture should include authentication and authorization controls, network access and inspection controls, and monitoring/enforcement controls for both the network and endpoints. No single technology currently will provide a full "zero trust" design and implementation—a combination of tools and services is needed to provide the full degree of necessary coverage. For most, a hybrid approach of both zero trust and existing infrastructure will need to coexist for some period of time, and emphasis should be placed on the common components and control categories that could suitably enable both, like identity and access management through directory service integration, endpoint security and policy enforcement, and network monitoring and traffic inspection. Zero trust projects are also long-term projects. In many cases, implementing a cohesive set of controls functioning together could take several years (perhaps as long as 4-5 years in some environments). As zero trust frameworks mature and evolve, so will standards and platform interoperability, likely facilitating more streamlined and effective approaches overall.

About the Author

Dave Shackleford, a SANS analyst, senior instructor, course author, GIAC technical director and member of the board of directors for the SANS Technology Institute, is the founder and principal consultant with Voodoo Security. He has consulted with hundreds of organizations in the areas of security, regulatory compliance, and network architecture and engineering. A VMware vExpert, Dave has extensive experience designing and configuring secure virtualized infrastructures. He previously worked as chief security officer for Configuresoft and CTO for the Center for Internet Security. Dave currently helps lead the Atlanta chapter of the Cloud Security Alliance.

Sponsor

SANS would like to thank this paper's sponsor: