4 ways to secure infrastructure and increase agility in a hybrid world

Table of contents

Protecting IT infrastructure such as data centers, networks and PCs has been a mainstay of security teams for decades. However, those times are changing, with a growing number of organizations embracing the cloud and new ways of working.

Many large IT organizations now operate in complex public cloud, multicloud and hybrid environments, with software development teams applying continuous integration/continuous delivery (CI/CD) techniques to rapidly push out new code. Meanwhile, the COVID-19 crisis shifted roughly half of most workforces from secure office environments to employees' homes, mingling personal devices and company assets on less secure home networks.

These same trends are driving digital transformation programs, but security typically is seen as a roadblock to IT modernization efforts. Market forces in 2020 have accelerated these demands as organizations seek ways to create more agile and resilient operations to keep pace, leaving security organizations faced with growing challenges such as lack of coverage for increased threats on the perimeter, poor visibility and control, manual processes, and increased restraints on resources.

In the face of these demands, security organizations need to quickly adapt their approaches to securing infrastructure in this fast-changing hybrid world. This paper explores four ways security can respond with new models and practices that are better aligned with today's business needs:

1. **SaaS-enabled security.** Delivering security with cloud connectivity, scale and efficiencies

2. **Extended detection and response (XDR).** Providing greater visibility and control into more environments

3. **Secure access service edge (SASE).** Enabling location-independent access to corporate resources, where access is controlled on the basis of extensive information about the context of the connection, the activity and the data being accessed

4. **Container security.** Providing a rapidly patched and consistently configured platform for microservices

1. SaaS-enabled security

An increasing number of security controls can be consumed as software as a service (SaaS), often referred to as security as a service (SECaaS). Unlike remotely operated security services, such as remote management of security hardware on-premises, modern SECaaS encompasses the architectural shift to security management, data storage, analysis and user interface components operated within the cloud.

Most traditional security vendors now offer access to their products in the cloud. Components such as vulnerability scanners, log collectors and agents continue to be deployed on site, but the functions of central management, configuration, data concentration, analysis and reporting are delivered by the vendor from a cloud-based environment. Onsite components, as well as those deployed in the cloud, communicate with the vendor's API over the internet using secure protocols, allowing a unified view of hybrid environments. System management operations are delivered by the vendor as a standard part of the service. SECaaS can therefore lift the system administration burden from the security team. Examples of this approach include security information and event management (SIEM), endpoint detection and response (EDR), and vulnerability testing **(see Figure 1**).

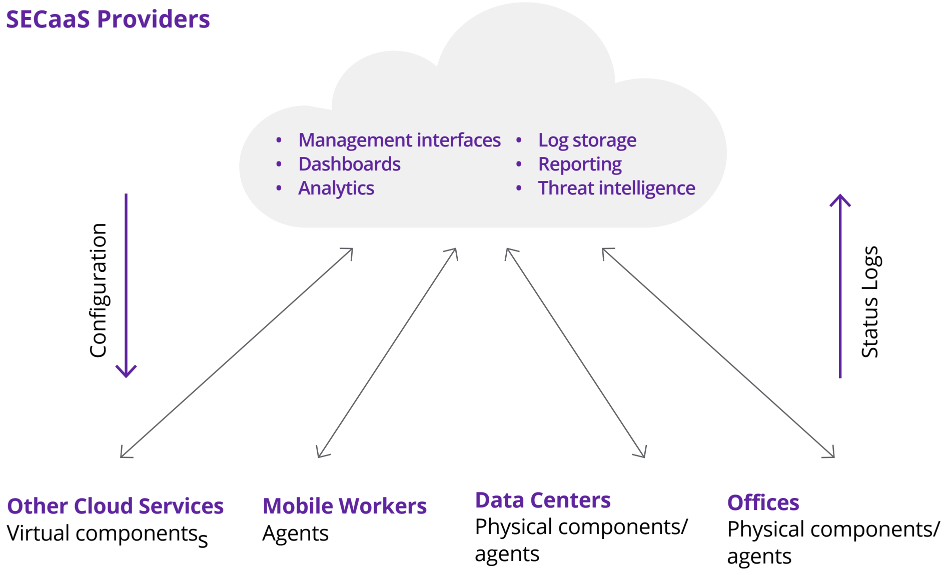

Figure 1. Typical SECaaS models secure multiple environments.

SECaaS solutions use APIs extensively, supporting links with other tools such as security orchestration, automation and response (SOAR). Integration with virtualized infrastructure and cloud platforms allows workloads to be monitored by the security service to identify unprotected instances and automate remediation by instructing the cloud service to deploy an agent or isolate threats until action is taken. This ensures always-on security, enabling protection from the moment the workload is functional.

Security organizations need to quickly adapt their approaches to securing infrastructure in this fast-changing hybrid world.

API-based communication between the agents and the SECaaS management layer enables the automated upgrade of agents, further reducing demands on the security team. Many SECaaS solutions include multiple protection capabilities within one deployable component, minimizing the costs of deploying multiple agents.

These solutions are typically designed for the cloud, containers and microservices, thereby enabling traditional and more agile deployment models to be protected by the same solution with common policies and reporting. Consolidation of tools reduces complexity, increases coverage, drives consistency, and simplifies deployment, maintenance and operation, enabling the security team to focus on key initiatives.

Despite the potential advantages, SECaaS deployment can face challenges in the shape of product maturity and dependence on the service provider. Cloud-managed solutions do not have the track record of on-premises solutions and may have fewer features, particularly compared with more mature on-premises solutions from the same vendor.

In addition, the SECaaS model introduces a third party, the service operation provider, with access to configuration details and systems logs. That data and the staff accessing it may be based in locations outside an organization's data sovereignty limits. The identities of the service administrators and their activities will be opaque to the client, as will much of the day-to-day operation, backups, business continuity, etc. In addition, the management portal is necessarily exposed to the internet, and platforms are frequently operated in leveraged, multicustomer environments. This situation can upset traditional security and compliance approaches, pitting visibility and direct control of operations against responsibility for delivering and resourcing operations. The service provider needs to provide visibility of internal controls together with security attestations and evidence of audited compliance with industry standards. However, the customer that licenses the solution is still accountable for security. Therefore, customers should apply appropriate due diligence during the purchasing process and hold regular compliance reviews with the vendor.

2. Extended detection and response

It is widely accepted that absolute prevention of security incidents is impractical; therefore, security teams must ensure detection and active response. To succeed, they need tooling to collect detailed information, transport it back to a central console, support incident analysis with interactive interrogation of the state of devices, and ultimately change the state of systems to disrupt attacks. Although not yet universally deployed, endpoint detection and response (EDR) tools are an effective, well-established approach to system defense and are a significant part of the toolkit for complex investigations and responses.

Consolidation of tools reduces complexity, increases coverage, drives consistency, and simplifies deployment, maintenance and operation, enabling the security team to focus on key initiatives.

EDR tools do have shortcomings in terms of visibility. EDR is typically deployed only on corporate-supplied end user computing devices. Attackers, however, may focus on more vulnerable areas, such as network-attached devices (storage systems, printers and network appliances), bring-your-own-device (BYOD) systems, cloud-based systems, and the cluster of internet of things (IoT) devices now connected to networks. Furthermore, other attack vectors such as email are not visible at the operating system interface where EDR normally resides. The level of monitoring on such systems, based on the logs they generate, is typically not as detailed as EDR capabilities.

Gaps in visibility may be addressed by introducing detection and response technologies specific to each area, with each tool providing detailed recording, interrogation and modification of state. For example, network detection and response provides network traffic analysis and integration with network control devices for enforcement of response. Unfortunately, multiple independent, specialist technologies are difficult to effectively manage. This creates silos of visibility and control, with the potential for attackers to hide in the cracks between these systems.

Ideally, a universal detection and response capability that not only encompasses different zones, including endpoint, network and cloud, but also unifies the information would allow analysts to trace activities that pass between zones and provide a single point of control for response. This concentration of visibility and control also provides a platform for analysis, artificial intelligence and automation.

Security vendors have developed the term XDR (extended detection and response) to describe such a universal, integrated DR tool, with “X” referring to “extended” or “everywhere” coverage. For the moment, there is no standard definition of XDR, what should be included in an XDR solution, and how much emphasis it should place on log ingestion, threat intelligence, analysis and automation. XDR as a concept risks being devalued as a solution when marketing teams tend to apply it to anything; however, the general consensus is that an XDR system should break down silos of information and enable effective response capabilities. The comparative immaturity of XDR is one of the challenges to its adoption. The utopian vision of universal detection and response isn’t likely to be achieved in a single implementation. Practical considerations may prevent the result from being truly universal, though the goal of increasing detection and response capabilities is still worthwhile.

Other complexities exist in ensuring that the organization has the analyst resources and expertise to respond to the insights provided. To effectively use the information, security teams must have detailed knowledge of the behavior of the systems and possible attack techniques. Machine learning has some promise in this area and is a capability increasingly used to supplement and even replace signature-based approach detections. It is also necessary to have a plan and appropriate authorizations for the response actions that may be required.

Unifying detection and response tends to promote the choice of a single vendor for all components, but security organizations are often faced with complex choices between best-of-breed and integrated suites of tools, vendor selection and management, and transition from existing tools.

XDR requires a clear long-term strategy. Organizations must choose technologies, vendors and service partners with long-term plans. They can start by focusing on the most mature elements of XDR, the highest risk elements to be covered, and the places where existing tools give the least useful insight. That typically leads to an initial concentration on endpoints. Once EDR is operating effectively within the limits of its visibility, it should be expanded into XDR.

3. Secure access service edge

Traditional security controls tend to be concentrated around the concepts of trusted and untrusted zones and systems, enforced by a strongly defined perimeter. Access tends to be granted based on trusted users, with minimal review of activity after network access is granted. Adoption of a fundamentally different approach to the corporate security architecture is a foundational step for a transformed business. This should be influenced by the industry-recognized concept of zero trust, whereby trust is applied to individual resources rather than entire networks. The SASE architecture enables a cloud-delivered, service-based and location-independent point of presence to enable secure access for distributed applications, services, users and devices. It can protect both traditional and digitally transformed environments for users wherever they are in the world (see **Figure 2**).

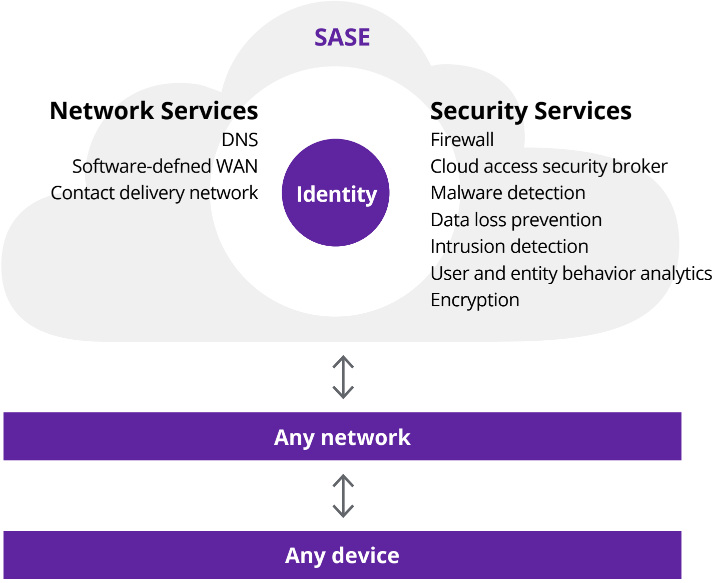

Figure 2. SASE architecture enables a cloud-delivered, service-based and location-independent point of presence.

Organizations must choose technologies, vendors and service partners with long-term plans. They can start by focusing on the most mature elements of XDR, the highest risk elements to be covered, and the places where existing tools give the least useful insight.

SASE replaces the concept of restricted points of access on a corporate network perimeter with a virtual perimeter for corporate and external users, with access decisions based on policy, identity, device and data-aware security controls. Instead of basing access decisions on the initiation of an external network tunnel to a corporate trusted network — making use of a user-device combination and allowing access to all internal resources — access decisions are based on individual connections, determined at the time of use. Decisions depend on identity, device profile, time, user, target context and location — who, where, what, when and how. Attackers may gain access to a device in a network, but they will be less able to move laterally when every action and access is challenged and reassessed.

There are different approaches to SASE. Some involve endpoint agents; some make use of software-defined wide area networking techniques; others are more closely aligned with content delivery networks. SASE architecture combines multiple controls, such as decrypted content inspection, sandboxing, filtering, credential theft prevention and data loss prevention plus context-sensitive authentication and authorization. These controls are combined into one coherent system via a uniform, centralized policy. SASE event logs can be combined with advanced behavioral detection and response capabilities such as XDR to enable end-to-end monitoring for all activity across the architecture, amplifying an organization's abilities to detect and block attacks as they are taking place. SASE architectures are typically hosted in the cloud, making use of the SECaaS approach and reaping the benefits arising from it.

SASE is not a small-scale, single-product deployment. It requires a strategic approach, overhauling traditional architectures and investments, replacing conventional internet service provider connections and virtual cloud-based wide area networks, such as MPLS, with a cloud-based backbone and SD-WAN. A shift to firewall as a service (FWaaS) is another element typical of this strategy, with hardware and on-premises perimeter firewalls moved onto a cloud backbone.

The concentration of security controls and policies into the SASE architecture, with tight integration between components, may trade off the ability to integrate against the performance of individual components. It may not be practical to adopt a best-of-breed approach across the entire architecture. However, the pace at which mainstream vendors are adopting SASE and the emergence of innovators mean that individual functional constraints should be short-lived and outweighed by the broader benefits that can be achieved by the architecture as a whole.

4. Container security

During the move to remote working in 2020, security teams adapted existing perimeter controls out of necessity and scrambled to implement or increase capacity for remote access, relaxing requirements and controls.

One trend in current application development is segmentation of the application into a collection of microservices in order to manage the complexities and dependencies of monolithic applications. This approach also has the desirable security property of failure isolation. The architecture coupled with the deployment technique of continuous integration and CI/CD sees the microservices deployed in containers that encapsulate all the code and dependencies into a single image.

These images go through frequent development and deployment iterations. While simplifying the application structure, container environments are themselves complex, with multiple layers of abstraction (images, containers, hosts, container runtime, registries and orchestrator) that require specialized tools to interpret, monitor and secure.

Immature deployments of microservice architectures can suffer from some of the same security issues as monolithic applications. For example, the incorporation of obsolete, unpatched, vulnerable components and libraries or misconfiguration can create flaws on multiple containers within the application. The rapid development and deployment of applications typically takes days rather than months, but it often conflicts with traditional security approaches such as code review, penetration tests and extensive change sign-offs.

Security organizations must ensure that security is built into the CI/CD process and supports rapid development cycles. This requires integration with the deployment tools and reaching beyond the containers themselves into the entire container life cycle. Doing this, making use of some of the inherent properties of containers, and rapid deployment can lead to more robust security. For example, making use of the automated testing and deployment inherent in CI/CD can support far more timely deployment of patches with a higher confidence that they have been tested thoroughly, ensuring a simple route to backing out any changes. Best practices include:

- **Securing the source.** A CI/CD process is an attractive way for attackers to deploy code, since if they can modify the source, CI/CD will see it delivered into production. Consequently, the components from which containers are constructed need to come from trusted sources with strong access controls and integrity checks on the components.
- **Incorporation of the latest versions.** Use of older, vulnerable components in container images is one of the leading causes for container vulnerabilities. The system needs to check for and incorporate the latest versions.
- **Continuous vulnerability scanning.** Incorporation of vulnerability scans into the CI/CD pipeline can add confidence that the latest components have been incorporated and configured correctly.

One trend in current application development is segmentation of the application into a collection of microservices in order to manage the complexities and dependencies of monolithic applications.

- **Immutability.** This approach says that no changes to running containers are acceptable. This requires updates to be made only through the tracked, versioned deployment process, ensuring consistency and repeatability. It also means changes on running containers are inherently suspicious.
- **Host security.** Use of stripped down, container-specific operating systems plus the use of and testing for best-practices benchmarks minimizes the attack surface.
- **Secrets management.** Containers do require secrets like API keys, credentials and certificates for authentication and authorization activities. This sensitive information should not be stored within container sources. Instead, it should be held in a separate repository and made available only on a need-to-know basis to the right containers and applications.
- **Monitoring and response.** Monitoring can be performed both within the container and by the underlying operating system. The concentration of containers on a single, simple task can assist with behavioral monitoring. The container-based approach also can assist in responses by blocking connections or pausing a single component.

Challenges also arise in managing containers. The number of containers that can exist and the speed at which they can be created and retired can be significant challenges for systems trying to monitor them and for analysts trying to interpret the behavior of the system.

While simplifying the development task by dividing it into a set of well-defined microservices may well improve the security of the application layer, the complexity of the infrastructure required to support that architecture at scale can be significant. For example, secret management platforms are difficult to integrate while accommodating needs for speed and scalability when there are hundreds or even thousands of microservices.

Organizational issues need to be addressed as the security function shifts from being a separate operation working at extended time frames on large, infrequent deployments to one closely integrated with, or delegated to, the development and deployment operation, which requires automation of processes and rapid decisions.

Containerization done well can improve security, but it does increase infrastructure complexity and so will require strategy, care, attention and resources.

Putting the security controls to work

During the move to remote working in 2020, security teams adapted existing perimeter controls out of necessity and scrambled to implement or increase capacity for remote access, relaxing requirements and controls. With workers using their own devices in shared home environments across public networks, attackers have been attempting to exploit these changes with themed phishing attacks to steal data and extort organizations with ransomware. Here's how these security strategies can help address these vulnerabilities:

- **SaaS-enabled security.** Aims to allow the same level of access and security whether you are working inside or outside an organization's traditional perimeter. It can also provide much more nuanced, fine-grained authorization decisions and content inspection than is possible with VPN technologies.
- **Secure access service edge.** Supports the security team, which is also working from home, by giving them access to security management tools that are designed to be operated successfully across the internet. This cloud-native approach enables security teams to manage tools remotely and support numerous changes in systems and applications.
- **Extended detection and response.** Expands visibility across the environment from endpoints, through networks to cloud systems. This approach also plays a part in efficiently capturing and responding without the need for physical intervention. Even basic endpoint detection and response has a role in remotely investigating and responding to attacks.

Changes in 2020 also brought on unprecedented demand for rapid application deployment across nearly all use cases, including online videoconferencing, health services, government aid programs, financial services, and online ordering and delivery.

Cloud provision of microservices has underpinned much of the expansion toward rapid development and CI/CD. Consequently, containers and all aspects of security around them are extremely relevant.

SECaaS is a key component of cloud provisioning for these new applications. In addition, XDR can assist in securing the endpoints used by developers and administrators and provide monitoring, investigation and response capabilities to infrastructure as it is adapted and rolled out. SASE also can play a role in effectively controlling access to new systems.

Security organizations must ensure that security is built into the CI/CD process and supports rapid development cycles.

Conclusion

Ongoing changes to infrastructure delivery techniques are supporting digital change and meeting demands for flexible, agile systems delivery and changes to usage patterns. These trends are requiring changes in the way security controls are implemented. At the same time, organizations struggle with the amount of information they already receive and are expending significant resources on maintaining the existing systems.

Four key technologies can strengthen an organization's security strategy to address the challenges, support transformation and build a platform for future developments:

- **SaaS-enabled security** to give a view across conventional organizational perimeters and to transfer the burden of security management tool maintenance to a service provider
- **Extended detection and response (XDR)** to provide more detailed visibility into and control of environments
- **Secure access service edge (SASE)** to consolidate existing controls and enable location-independent access to resources, with fine-grained control of the access provided in support of a zero trust approach
- **Container security** to provide a rapidly patched and consistently configured platform for microservices

Adopting these wide-ranging approaches helps organizations' security teams to increase coverage and visibility of security-related events, allowing them to concentrate on managing security outcomes instead of spending the time managing the platforms, which is time-consuming and distracts them from their core job.

About DXC in security

Recognized as a leader in security services, DXC Technology helps customers prevent potential attack pathways, reduce cyber risk, and improve threat detection and incident response. Our expert advisory services and 24x7 managed security services are backed by 3,000 experts and a global network of Security Operations Centers. DXC provides solutions tailored to our customers' diverse security needs, with areas of specialization in cyber defense, digital identity, secured infrastructure and data protection.

Learn how DXC can help protect your enterprise in the midst of large-scale digital change at **www.dxc.technology/security.**

About the authors

Dr. Rhodri Davies is DXC Technology's security and service operations architect, with over 20 years of experience in the field. He works in the Managed Security Services division of DXC, where he concentrates on the technologies required to secure DXC's customers and the way those technologies are operated day-to-day to provide an effective service.

Mike Dutton, senior security architect for DXC Managed Security Services in Australia, has almost 10 years of experience at DXC and has been involved in almost all of DXC's managed security services, with a focus on cloud security. He ensures that DXC chooses appropriate and effective security controls to support customers in achieving their digital transformation goals.

Yahya Kharraz is an information security architect at DXC Technology with a wide range of experience in security, from technical solutions to security governance and risks. He has solid experience in endpoint and infrastructure security and is passionate about the cloud, web application development, automation and coding. As part of his professional interest, he is constantly exploring and evaluating cutting-edge technologies.

Dirk Thys is currently a security compliance advisor with DXC Technology. He is also the lead architect/engineer responsible for the hardening and security of Platform DXCTM worldwide. Dirk works closely with partners and vendors to achieve DXC business goals and clients' requirements. He has held various IT operations, engineering, project and program lead roles, and managed platform security teams with responsibility for delivering compliance management services for 600+ clients worldwide.

Learn more at
dxc.com/technology/security

Get the insights that matter.
dxc.com/optin

About DXC Technology

DXC Technology (NYSE: DXC) helps global companies run their mission critical systems and operations while modernizing IT, optimizing data architectures, and ensuring security and scalability across public, private and hybrid clouds. The world's largest companies and public sector organizations trust DXC to deploy services across the Enterprise Technology Stack to drive new levels of performance, competitiveness, and customer experience. Learn more about how we deliver excellence for our customers and colleagues at **DXC.com**.