WP.01/

VirnetX Secure DNS & Zero Trust Architecture in VirnetX One Matrix

This is a technical description of the VirnetX One Matrix product. VirnetX SDNS technology requires knowledge of networking and security on the part of the reader.

This paper describes VirnetX's Secure Domain Name System (SDNS) technology. Next, it covers Zero Trust and how VirnetX One Matrix implements Zero Trust with its SDNS technology. Finally, it describes where you can improve your network security by implementing VirnetX One Matrix to protect your applications, services, devices, and controllers.

Secure Domain System (SDNS)

The domain name system protocol was developed in the early 1980s when the ARPA Internet was first created (https://www.rfc-editor.org/rfc/rfc882). When the Internet was first created, it was very small, and more importantly, the network was trusted; security was never a significant consideration in its design.

The functionality of Domain Name Service (DNS) is simple. The DNS protocol translates the resource's name on the Internet to a numeric IP address because users of the Internet can't be expected to know the addresses of all resources – there are billions of them!

Internet Routers route requests based on a unique numeric address for each device/endpoint called Internet Protocol Addresses, an example of an IP version 4 (or IPv4) address is 10.10.10.10. An IPv4 address is four numbers between 0 and 255 separated by dots. Later, IPv6 was created, which supports much longer numeric addresses because we're running out of IPv4 addresses, but for compatibility reasons, all important services are still reachable with IPv4 addresses.

A few key attributes of the original DNS design, which are features of the DNS today, include:

- The requester of the DNS answer can be anonymous.
- Generally, all anonymous users can get the IP address associated with a name and get the same DNS answer in most configurations.
- Because the IP address determines how traffic is routed, it tells with some accuracy the location of the endpoint, so if anyone can get that IP address from the name, anyone can attack that specific endpoint. These attacks can include denial of service (DoS) attacks, surveillance attacks to gain information about your service, scanning the IP address for signatures that reveal the specific applications and operating system variants running at the IP address, and finally, attacks to exploit vulnerabilities in the OS/application running at that IP address.

The VirnetX SDNS technology used by VirnetX One and VirnetX Matrix is an improved DNS system with significant security advantages over the original DNS design and the revised DNSSEC design.

The VirnetX SDNS technology has the following attributes:

- All requestors of SDNS must be cryptographically authenticated.
- To receive a DNS response other than "name not found," the requestor must be in the security policy of the requested endpoint (this security policy is defined by the owner of the endpoint).
- As part of responding with an IP address, the VirnetX SDNS technology dynamically creates a secure VPN tunnel between the requestor and responder hosts. SDNS responds to the DNS request with a secure IP address that automatically routes traffic through an end-to-end encrypted VPN tunnel to the host at the target name.
- In the 1990s, security issues found in DNS led to improvements to the original DNS design called DNSSEC (or DNS security). VirnetX SDNS is different from DNSSEC. DNSSEC cryptographically protects the DNS answers to prevent DNS cache poisoning attacks. VirnetX SDNS has the same features as DNSSEC (cryptographically protecting DNS answers) but goes beyond DNSSEC to also require cryptographic authentication (no anonymous requestors), and to put a secure communications channel in place based on the defined security policy as a part of responding to the original DNS request.

WP.02/

VirnetX Secure DNS & Zero Trust Architecture in VirnetX One Matrix

Almost all network applications and VPN servers protecting network application services require an open listening port on the firewall (usually a demarcation point between public and private networks) to facilitate a network socket connection into that network application service that is typically running on the private network.

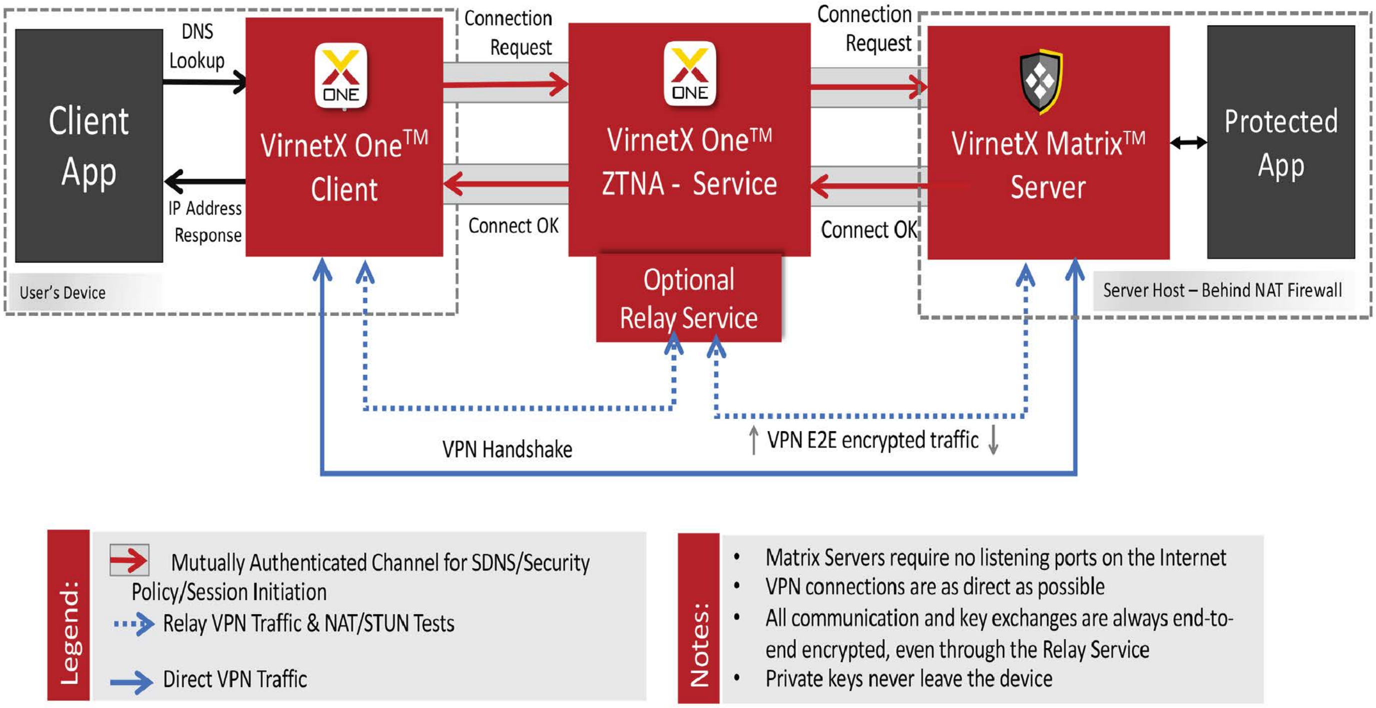
A unique feature of the VirnetX SDNS technology is that no open ports are required for an application protected by VirnetX SDNS. Both the requestor and responder endpoints can be on private networks behind network address translation (NAT) firewalls \ gateway \ and routers, and the VirnetX SDNS technology will still dynamically create a secure tunnel between the endpoints and return a secure IP address for communication over that VPN tunnel.

The way that the VirnetX SDNS technology accomplishes the VPN connection without any open ports is it uses the authenticated network connection to the SDNS service to facilitate the STUN (session traversal utilities for NAT) and TURN (traversal using relay around NAT) protocols to connect the VPN when no direct network socket connection seems possible.

The encryption channel of the VPN is always end-to-end between the two endpoints; however, the network connection is either direct using STUN if possible or through a network relay which reflects the traffic without decryption. Once the VPN is in place and the unknowable secure IP which routes traffic through the VPN is returned to the requesting application, the requesting application can then connect a network socket through the VPN to the requested endpoint (and only inside the VPN is that listening port visible).

Figure 1 depicts the VirnetX SDNS Technology implemented by VirnetX One Matrix. The secured applications' DNS lookups are seen by the VirnetX One Client. If the request is for a secure domain name, the connect/DNS lookup request is forwarded to the VirnetX One ZTNA Service. The requester is securely authenticated to the VirnetX One ZTNA service. If the requestor is in the security policy of the requested Secure Domain Name, a dynamic VPN is put in place between the requester and responder using STUN/TURN if required, and the secure IP address is returned, which will route the traffic through the VPN. This secure IP address is returned to the requesting app, which will then route all its traffic through the VPN.

Figure 1 - Architecure of VirnetX SDNS Technology



Legend:

- Mutually Authenticated Channel for SDNS/Security Policy/Session Initiation
- Relay VPN Traffic & NAT/STUN Tests
- Direct VPN Traffic

Notes:

- Matrix Servers require no listening ports on the Internet
- VPN connections are as direct as possible
- All communication and key exchanges are always end-to-end encrypted, even through the Relay Service
- Private keys never leave the device

WP.03/

VirnetX Secure DNS & Zero Trust Architecture in VirnetX One Matrix

Zero Trust Architecture

The Zero Trust architecture is well defined in the NIST 800-207 publication (see https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf).

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

The key tenets of Zero Trust are defined in section 2.1:

- All data sources and computing services are considered resources.
- All communications are secured regardless of network location.
- Access to individual enterprise resources is granted on a per-session basis.
- Access to resources is determined by dynamic policy – including the observable state of client identity, application/service, and requesting asset – and may include other behavioral and environmental attributes.
- The enterprise monitors and measures the integrity and security posture of all owned and associated assets.
- All resource authentication and authorization are dynamic and strictly enforced before access is allowed.
- The enterprise collects as much information as possible about the current state of assets, network infrastructure and communications and uses it to improve its security posture.

Section 2.2 goes on to define Zero Trust at the network level in more detail:

- The entire enterprise private network is not considered an implicit trust zone.
- Devices on the network may not be owned or configurable by the enterprise.
- No resource is inherently trusted.
- Not all enterprise resources are on enterprise-owned infrastructure.
- Remote enterprise subjects and assets cannot fully trust their local network connections.
- Assets and workflows moving between enterprise and non-enterprise infrastructure should have a consistent security policy and posture.

Implementing Zero Trust with SDNS Technology in the VirnetX Matrix

Now that you understand the VirnetX SDNS technology and Zero Trust principles, let's discuss how the VirnetX One Matrix utilizes VirnetX SDNS to implement Zero Trust Networking:

- VirnetX One Matrix secures all network communications behind secure domain names regardless of the network location.
- All usage of VirnetX One Matrix requires authentication of the endpoint devices and users of those devices.
- VirnetX One Matrix allows organizations to define rich security policies for granting access to secured applications.
- In addition to putting a dynamic encrypted tunnel in place for all communications to a secured application. The organization admin specifically limits the network ports available to the accessing devices providing a high degree of network segmentation to only what is needed for access to a specific application (User & Device Isolation)
- VirnetX One Matrix has features that allow the organization's network administration to monitor endpoint devices and network utilization to ensure the integrity and security posture of assets.
- VirnetX One Matrix collects information on devices and network communications to improve the security posture.
- Because VirnetX One Matrix encrypts end-to-end all communications from clients to servers regardless of their location on the network, there is implicitly no trust in the underlying local network or any network or underlying infrastructure.
- VirnetX One Matrix integrates SDNS with Zero Trust principles (the most basic underlying protocol which starts all communication on the Internet).

WP.04/

VirnetX Secure DNS & Zero Trust Architecture in VirnetX One Matrix

How Does VirnetX One Matrix Implement SDNS Zero Trust to Keep You Safe

Let's discuss external network threats. We've all received automated auto-dialer marketing phone calls. A similar approach is used by hackers to automate known vulnerabilities on servers. Applications running on HTTP (TCP port 80) or HTTPS (TCP port 443) will commonly see repeated attempts to exploit known vulnerabilities, shell shock, Log4Shell (and many others). SSH servers running (TCP 22 or another non-standard port) and RDP servers (TCP port 3389) will see attempts to log in with common usernames/passwords.

When VirnetX One Matrix is placed in front of these apps, the listening IP addresses and ports are no longer reachable from the Internet by third parties. They can only be reached by VirnetX One Matrix authenticated users through a VPN tunnel. So, these applications become "invisible" on the Internet and are not attackable by automated attacks.

A more dangerous attack is an adversary who is not running a robot attack but is specifically targeting your application knowing the domain name associated with your application. In this case the adversary will receive the response "name not found" when they attempt to get the IPv4 address associated with your application. They don't know if your application is available on the Internet or where it is hosted, in fact they see no evidence that the application even exists.

With external attacks blocked the attackers only way to attack your application is to compromise one of your legitimate users and attack the application through their authenticated device. In this case VirnetX One Matrix is tracking user device and network communications attributes to also block the compromised insider attack of the application.

Where Can You Deploy VirnetX One Matrix SDNS Zero Trust

Most organizations have applications that fit into three categories:

1. Publicly Available Applications: Your organization needs to have publicly available on the Internet for reaching new customers (i.e., your organization's public website).
2. 3rd party SaaS Applications (Microsoft 365, Google Workspace, DropBox, SalesForce, etc.)
3. Self-Hosted Applications that are either on premise or on your own Virtual Private Cloud (VPC).

VirnetX secures Self-Hosted Applications. Examples of common applications VirnetX protects include:

1. Private File Servers
2. Private Web Servers \ Application Servers
3. Database Servers
4. SSH (Secure Shell)
5. RDP (Remote Desktop)
6. VNC (Virutal Network Server)
7. DevOps Systems (Environments used by developers to develop, test and deploy software products)
8. Internal Messaging Servers (Chat, SMS)
9. Mail \ Groupware Servers (POP3/IMAP/SMTP/Exchange Server)
10. Security Camera Concentrators
11. Bastion Hosts (Hardened Command and Control Systems)

VirnetX One and Matrix are an important layer for all applications that are considered highly sensitive and require additional protection from your security team. VirnetX does not eliminate the need for baseline security tools such as endpoint device protection, security patch management, identity management, application firewalls, and security incident event management (SIEM) that protect all your networks. VirnetX integrates seamlessly into your existing security infrastructure and enables the highest possible level of security for your most sensitive applications.