Security Implications of IPv6

Michael H. Warfield
Senior Researcher and Fellow
X-Force
Internet Security Systems

Executive Summary

Internet Protocol version 6 (IPv6) contains numerous features that make it attractive from a security standpoint. It is reliable and easy to set up, with automatic configuration. Huge, sparsely populated address spaces render it highly resistant to malicious scans and inhospitable to automated, scanning and self-propagating worms and hybrid threats.

IPv6 is not a panacea for security, though, because few security problems derive solely from the IP layer in the network model. For example, IPv6 does not protect against misconfigured servers, poorly designed applications, or poorly protected sites. In addition, IPv6 and IPv6 transitional mechanisms introduce new, not widely understood, tools and techniques that intruders can use to secure unauthorized activity from detection. These IPv6-derived efforts are often successful even against existing IPv4 networks.

Since many network administrators have yet to take advantage of IPv6, they may be unaware of IPv6 traffic that has tunneled into their networks. Attackers are already using this potential oversight to establish safe havens for attack.

Fortunately, existing protection technology is equipped for IPv6, making protection across this emerging standard both practical and straightforward. This whitepaper discusses the security implications of IPv6 and solutions that enable administrators to protect against attacks, intrusions and backdoors that take specific advantage of the protocol.

Introduction

Internet Protocol version 6 (IPv6), under development for many years, is already deployed extensively on production networks. Yet many network administrators feel they do not need to worry about IPv6 as long as IP version 4 (IPv4) suits their current needs. The irony is that IPv6 is readily available to anyone with an IPv4 address. Most Internet-enabled platforms are already IPv6-ready and only need simple commands to fully utilize the protocol - even without corresponding infrastructure support.

Organizations unprepared to support or recognize IPv6 are unlikely to defend themselves against IPv6-enhanced attacks. These facts have not gone unnoticed by the darker elements of the Internet underground. Hackers are already actively taking advantage of new IPv6 services, and turning this lack of understanding about IPv6 to their own advantage.

A Brief Overview of IPv6

The remarkable growth of the Internet Protocol version 4 (IPv4)-based Internet has highlighted several fundamental limitations with that protocol. Internet Protocol version 6 (IPv6) addresses these issues and provides additional enhanced services and functionality. IPv6, also called IP-NG, is the "next generation" Internet Protocol and is the designated successor to IPv4.

Although some aspects of IPv6 are still under development, the basic protocols, conventions, and formats have been stable for years and enjoy wide support. Real-world production deployment (allocation and assignment of production network addresses or prefixes) has been underway for several years, and IPv6 is no longer considered experimental.

The most commonly discussed concern with IPv4 is the perception that IPv4 provides an insufficient number of individual addresses to meet future needs. While conservation, recovery, and techniques such as Network Address Translation (NAT) have improved address availability and prolonged the longevity of the IPv4 address pool, there remains a limit to the future growth of IPv4 due to its 32 bit address fields. IPv6 dramatically increases this limit by expanding the number of bits in the address fields from 32 bits to 128 bits.

As IPv4 has expanded, IPv4 routing tables have expanded as well. This expansion has heavily taxed the underlying routing infrastructure. While techniques such as Classless Inter-Domain Routing (CIDR) and aggregation have slowed this growth, IPv4 use still expands faster than the capacity of the routing infrastructure. IPv6 deemphasizes growth impact by way of more formalized network and subnetwork boundaries and aggregation of smaller site networks into aggregation pools and aggregation IDs.

Much of the fragmentation of the IPv4 address space has been caused by the inherent difficulty in renumbering IPv4 networks. IPv6 addresses this limitation through transition mechanisms and auto-configuration methods that allow dynamic renumbering, multiple addresses, and transition periods which ease transitions between address prefixes.

IPv6 also improves on many of the security shortcomings that exist in IPv4. In particular, IPv6 contains many enhanced security features, such as IPSec (AH/ESP), that were back-ported into IPv4. Others, such as resistance to scanning, are only possible under the IPv6 addressing scheme. For example, the massive size of the IPv6 address space by itself creates significant barriers to comprehensive vulnerability scanning.

Other IPv6 features, such as the autoconfiguration of addresses, make it complicated for a malicious attacker to probe systems for weaknesses. These factors will not stop random or pseudo random scanning, but they will make it difficult to scan specific IPv6 networks. However, IPv6 networks can be scanned effectively if they are poorly designed (as in the traditional IPv4 model) and use dense address allocations and/or well-known addresses for services and routers.

Auto configuration makes IPv6 relatively easy to setup and renumber on demand. Consequently, it also makes it easy for an intruder who has already gained access to a local subnetwork to announce rogue routes and routers to further an attack, or to route multiple compromised systems through tunnels under illicit control.

Transition tunnels and tunneling routers make it possible to deploy islands of IPv6 support within a larger sea of IPv4 networks without having the IPv6 routers directly connected to each other -- or even requiring IPv6 routers at all. This arrangement allows intruders to subvert simple workstations and use them as routers to direct traffic across entire subnetworks without having to compromise infrastructure routers or firewalls.

The State of IPv6 Deployment and Availability

IPv6 deployment has been relatively slow in North America, which has IPv4 saturation. IPv6 has been much more popular in Europe and Asia, where IPv4 is less prevalent and where IPv6-only production networks already exist. An IPv6 ISP is expected to come online in Asia by the end of 2003, with many more to come that offer only IPv6 services. Australia largely follows the IPv4-rich model of North America in which IPv6 is not heavily deployed.

This deployment is analogous to the deployment of digital cellular technology. Analog cellular telephony networks were already extensively deployed in North America when digital cellular technology was first introduced, which slowed deployment of digital cellular services. The lack of a widely deployed analog infrastructure allowed Europe, Asia, and South America to aggressively deploy superior digital technology without the need to recoup an investment in an intervening analog step.

Much work has gone into developing standardized IPv6 transition mechanisms to ease the shift from IPv4 to IPv6. SIT ("Simple Internet Transition" or "Six In Tunnel"), 6to4 automatic SIT tunnels, and IPv6 over UDP are common examples of these technologies.

These transition mechanisms couple with well-connected and easily available tunnel brokers to make IPv6 readily available to anyone with an IPv4 address, regardless of whether IPv6 is supported on any given network. In North America, many administrators do not realize that IPv6 is available. These networks may already carry IPv6 traffic without administrator awareness.

Tunnel brokers provide IPv6 tunnels to clients across an intervening IPv4 network. Several tunnel brokers do so free of charge to promote the propagation and deployment of IPv6. Most providers require some form of sign-up procedure, especially for large network-size prefixes (/48 or /64 size networks and subnetworks). However, registration is rarely more complicated than an agreement to abide by an AUP (Acceptable Use Policy).

Tunnel brokers provide both Internet6 (address prefix 2001::/16) and 6Bone (address prefix 3ffe::/16) prefixes. Some tunnel brokers feature single address routing. Others provide /64 subnets. Other brokers, such as FreeNet6 (6Bone), provide entire /48 networks (65,536 subnets) to create an automatic tunnel broker service for clients that can dynamically adapt to changes in dynamic addresses such as DHCP addresses. Some brokers such as Hurricane Electric (Internet6) require changes to tunnel endpoints performed via their Web interface and do not adapt easily to dynamic address changes of the tunnel endpoints.

By comparison, 6to4 tunnels are automatically configured SIT tunnels based on the IPv4 address of the host. This option requires no tunnel broker and no support from the underlying IPv4 network beyond simple forwarding of IPv4 datagrams, and no blockage of IP protocol 41 (IPv6 on IPv4 - the SIT protocol). No AUPs are signed and no permissions required. Anyone with an IPv4 address can immediately be on IPv6 using 6to4 auto SIT tunnels with an entire /48 size IPv6 network at their disposal.

Network administrators managing IPv4 networks often overlook or ignore IPv6. They typically do not recognize its presence or its availability, and they frequently lack the skills or expertise to manage it. So they assume it is not present on their networks.

Unfortunately, this assumption is dangerously misguided. Thanks to the promotional efforts of various standards bodies and the transitional efforts of the Internet Engineering Task Force (IETF), IPv6 is available nearly anywhere IPv4 is available. Due to ignorance, lack of experience, and inertia, the security and administrative personnel tasked with defending IPv4 networks have not kept pace with the growth of IPv6.

The underground community of blackhats knows IPv6, and has developed the expertise to take advantage of it - especially given the relative lack of expertise on the part of the average network administrator. This expertise reflects a similar regional divide to the deployment of IPv6, with better IPv6 skills developing in parts of the world that are less rich in IPv4 technology.

Operating System Support

IPv6 is supported on most platforms and operating systems. It often only requires a simple command, configuration option, patch or upgrade to enable it.

Microsoft Windows

Recent versions of Microsoft Windows (Windows XP and Windows 2003 Server) have integrated IPv6 support. Windows XP only requires the command "ipv6 install" to enable native IPv6 on Ethernet interfaces. It also installs and enables support for SIT tunnels, 6to4 tunnels, and pseudo interfaces for Teredo IPv6 over UDP tunnels. The command requires rebooting the Windows system to enable IPv6, which disrupts any current activities or connections. The command enables 6to4 by default, with a "wellknown address" that can easily be scanned for against the IPv4 address.

Windows XP includes preliminary support for Teredo IPv6 over UDP tunnels. This feature, in theory, allows UDP tunnels to bypass firewalls that block SIT and 6to4 tunnels.

Windows 2000 supports IPv6 through a simple, free download from Microsoft. The update can be applied to any version of Windows 2000 SP1 or greater. While there are some limitations to this update, (it doesn't support DNS servers on IPv6 but does support IPv6 addresses from IPv4 DNS servers), it supports the full IPv6 stack and IPv6 applications. The update requires the Windows 2000 system to be rebooted in order to enable IPv6. Consequently, any active connections or system activity is disrupted by the installation of IPv6 on Windows 2000.

Older versions of Windows such as Windows NT, Windows 98, or Windows 95, have IPv6 support available from third party add-ons.

Linux

Many Linux distributions support IPv6 via simple activation and configuration. These systems do not have to be rebooted to activate IPv6. Recent versions of the Linux kernel include IPv6 firewall support. Most distributions provide protocol translation packages and proxies.

While details vary from distribution to distribution, RPM-based distributions such as Red Hat or Mandrake merely require the addition of the configuration variable, "NETWORKING_IPV6=yes," to the global network configuration file, "IPV6INIT=yes," to the individual interface file, and then a restart of the network subsystem. The IPv6 subsystem can also be manually started without a reset of the network subsystem. The system does not need to be rebooted in either case, nor are IPv4 connections affected by the IPv6 startup. If the Linux system is used as a router, "IPV6FORWARDING=yes" must be added to the global network configuration file.

By default, 6to4 is not enabled when activating IPv6 on Linux systems. This process requires the addition of a single additional parameter, "IPV6TO4INIT=yes," to the interface configuration file. Without any other configuration parameters, this action enables the 6to4 interface with a well-known address that can easily be scanned for against the IPv4 address.

Configuration details for Debian or Slackware distributions are significantly different than RPM-based distributions, but the manual commands are basically the same across all distributions of Linux.

UNIX
All recent BSD distributions (FreeBSD, NetBSD, OpenBSD, etc.) support IPv6 and IPv6 tunneling. Solaris and Solaris x86 fully support IPv6 in version 8 and higher. Solaris systems require only the creation of "hostname6.{interface}" files to enable IPv6 on the interface named {interface}. AIX and HP/UX also have full support for IPv6 in recent versions.

IPv6 configuration commands can be executed manually by anyone with administrative capability without requiring that the network or the system be reset. The IPv6 configuration can take place on a running UNIX system without disruption of system activities or any IPv4 connections.

Other
IPv6 support is available on Apple platforms beginning with Mac OSX. Major router manufacturers such as Cisco support IPv6 in recent router and router management software, including OSPF and BGP. Many implementations of PPP (Point to Point Protocol) now include support for IPv6. IPv6 may therefore be active anywhere PPP can be connected or tunneled, including over transporting VPNs, serial links and UDP tunnels. In summary, it is clear that IPv6 enjoys wide vendor support.

Notable Exceptions
Low-end broadband routers and DSL router consumer products which provide Network Address Translation (NAT) for connecting multiple IPv4 devices to a single DSL or Broadband address generally do not support IPv6. Most SIT tunnels, including 6to4 auto-tunnels, are incompatible with NAT. The only choices are for the NAT device itself to support IPv6 and SIT, or to use an IPv6 over UDP protocol such as Shipworm or Teredo, even though this functionality is still an evolving standard.

NAT devices based on Linux or BSD have no difficulty providing and supporting IPv6, but low-end devices based on a proprietary, embedded OS typically do not. It is the lack of support for IPv6 on these low-end devices that drives the development of transition mechanisms such as Teredo - and consequently risks additional security complications where IPv4 and IPv6 interact.

Windows XP includes preliminary support for Teredo and can access IPv6 over UDP. As the standards process stabilizes this protocol, support will appear on a wider variety of platforms. Teredo servers will deploy on the Internet. The risks from wide spread availability of IPv6 over UDP, therefore, are likely to increase. Many firewalls, even stateful firewalls, permit outbound UDP traffic. Widespread deployment of Teredo and similar protocols increase the risk of bypassing firewalls that have not been reconfigured to take into account IPv6 over UDP.

IPv6 and the Internet Underground

The Internet underground maintains IPv6 IRC sites and servers, IPv6 Web sites and IPv6 ftp sites, indicating that elite hackers and crackers have been on top of IPv6 for some time. Soon, even less sophisticated attackers and "script kiddies" will regularly communicate over IPv6 and use IPv6 against those unfamiliar with the next generation of Internet Protocol.

An example of a legitimate IPv6 IRC site can be found here:

http://www.hs247.com/ipv6ircservers.html.

Underground sites now offer IPv6-enabled and IPv6-specific tools such as relay6, 6tunnel, nt6tunnel, asybo, and 6to4DDoS. Relay6, 6tunnel, nt6tunnel, and asybo are protocol bouncers which accept connections on IPv4 or IPv6 and redirect those connections to IPv6 or IPv4. This ability allows IPv4-only applications to connect to IPv6 services and vice versa. While these tools are legitimate, they are easily abused by the underground to create tunnels and redirects for backdoors and trojans. By comparison, 6To4DDos is a Distributed Denial of Service attack tool specifically designed to attack IPv6 sites and to attack IPv4 sites by using 6to4 tunneling.

Even mainstream sites such as Freshmeat.net offer IPv6 tools such as halfscan6 and netcat6 which are useful to the underground community. These IPv6-enabled versions of established open source security tools are frequently used by defenders and attackers alike.

IPv6 patches have been released for many favorite underground trojans, backdoors, and zombies. IRC "bots" or "robots" such as Eggdrop have been adapted to utilized IPv6 IRC sites for command channels. Even without IPv6 patches, protocol bouncers enable IPv6 access to many older tools and exploits.

Backdoor programs can lurk on an IPv6 6to4 interface hidden on a system that otherwise has no IPv6 facility. An IPv6-based backdoor simply configures 6to4 on the compromised system and picks an SLA (Site Local Aggregation - the 16 bit IPv6 subnet number) and an EUI (End Unit Identifier - the lower 64 bits of the IPv6 6to4 address) and then listens on that specific backdoor address and port. This port does not show up in IPv4 security scans.

Even if the host is scanned for IPv6 6to4 access, the scanner must determine the exact SLA and EUI in order to begin a scan for the port on that device. To do so successfully is quite an achievement - analogous to guessing an 80 bit key just to get started. This information can be detected by properly configured intrusion detection systems (IDS) monitoring for backdoor traffic. In other words, if administrators know to look and know where to look, these backdoors can be detected.

Some operating systems allow applications to listen for IPv6-only traffic and do not require the application to listen to specific addresses to avoid detection through the IPv4 interfaces. Others, such as Linux, deliver IPv4 traffic to IPv6 applications as IPv6 traffic, utilizing IPv6 compatibility addresses (IPv6 addresses which logically equate to IPv4 addresses).

On platforms such as Linux, backdoors and trojans attempting to hide from detection by IPv4-based scanners must take the additional measure of only listening on specific IPv6 addresses and not the IPv6 "receive-any" address of "::". This modification is not difficult to do and works equally well on platforms with even stricter isolation between the two protocol stacks.

IPv6 addresses hidden behind an IPv4 interface create a form of stealth barrier to detection by many scanner technologies currently in use. Some forms may be detectable only by sophisticated host-based security scanners or IPv6-aware network IDS. The inherent difficulties in scanning address spaces as large as a /48 IPv6 network with 80 bits of host addressing, make the detection of stealth backdoors via scanning from the external network almost impossible. A fusion of IPv6-aware network scanning and IPv6-aware intrusion detection can alleviate the threat.

The same holds true for "reverse backdoors" - backdoors and trojans that connect outwards from a compromised host. These attack tools do not hide server ports behind 6to4 stealth interfaces but instead hide traffic in SIT tunnels or in UDP-based IPv6 tunnels.

Compromised hosts may advertise IPv6 routes and forward IPv6 traffic back through themselves for an entire network behind firewalls and NAT devices. Even devices on a private address space may become globally visible and routable over IPv6 through Shipworm or Teredo type IPv6 over UDP tunneling, bypassing the NAT devices and firewalls.

Evidence already exists that intruders use IPv6 as a screen to avoid detection. Break-ins against "honeypots" reflect clear evidence that the attacker enabled IPv6 over IPv4 to create communications tunnels that evade security scanning and IDS detection. Lance Spitzner made the announcement of such a capture on his honeypots mailing list in December of 2002:

http://lists.insecure.org/lists/honeypots/2002/Oct-Dec/0105.html

Malicious code often contains IPv6-capable components such as 6to4DDos and other DDos flood tools. An example of a hybrid worm with IPv6 capabilities was captured by the Honeynet project:

http://project.honeynet.org/scans/scan25/sol/NCSU/main.html

IPv6 backdoors, Trojan horse programs and assorted malicious code easily evade most IPv6-unaware security or vulnerability scanning programs. These attacks also easily evade most IDS systems that are not IPv6-aware. If an IDS only examines IPv4 traffic and doesn't support IPv6, either natively or over various tunneling and encapsulation schemes, then an intruder can easily deliver exploits through unsupported tunneling mechanisms.

An IDS must be able to decode IPv6 and IPv4 equally well to detect these exploits and backdoors. The IDS must dig deeper into the packet and analyze a deeper level of the encapsulated traffic to handle either static SIT or 6to4 auto SIT. To handle something like Teredo, the IDS must dig even deeper into UDP than the current practice of un-encapsulating the IPv6 traffic.

In summary, the "black hats" often have deeper expertise and better tools than many "white hats" and security professionals trying to protect their networks. And that makes for a very dangerous situation in which IPv6 knowledge becomes increasingly critical over time.

A Look at IPv6 Addressing and Standards

A basic understanding of IPv6 addressing, its structure, and how it was defined is an important foundation for identifying the sources of the risks associated with IPv6 and its users and abusers.

IPv6 addresses look more intimidating than IPv4 addresses. Instead of four "octets" (number 0-255) separated by dots (a dotted quad), IPv6 addresses are a series of 16 bit hex numbers (number 0-ffff) separated by colons. There can be up to 8 of these numbers representing a single IPv6 address. There is, additionally, a shorthand notation in which "::" represents any number of "zero words."

Examples:

IPv4: 192.168.16.131
IPv6 (no zero words): 2001:470:104:20:202:b3ff:fead:42ba
IPv6 (with zero words): 2001:470:104:20:0:0:0:1
IPv6 (with zero shorthand): 2001:470:104:20::1 (previous example :0:0:0: = ::)

IPv6 addresses are well structured and can actually be simpler to understand than IPv4 addresses, which may have arbitrary CIDR (Classless Inter-Domain Routing) network boundaries and subnets, and similar complications.

For general-purpose use, the upper 16 bits (the first word or hex number) define certain standard "/16" (16 bit netmask) prefixes, each of which may have its own internal convention. Not all are routable. While the exact definitions and divisions vary, the first word in the address usually determines the category and scope under which the address falls. Of the six commonly encountered top prefixes, only three are of "global" scope, meaning that they can be routed on the Internet. The other three are more restricted scopes and are only of concern within the local network environment, much like private addresses on IPv4.

The entire top-level range of 2000::/3 (2000::/16 through 3fff::/16) is meant to be allocated for globally routable addresses. This range currently contains several prefix allocations, all of which are routable, with different purposes, allocation schemes, and formats. The 2001::/16 prefix is allocated for production IPv6 Internet assignments, while 3ffe::/16 is currently allocated for the 6Bone, the experimental test bed. The 2002::/16 prefix is used for 6to4 SIT autotunnels, described below. On networks that do not intend to support IPv6, a Network IDS can be configured to detect IP traffic with a version number 6 in the IP header. This presence could be indicative of malicious, or at least, non-supported network traffic if IPv6 is unsupported and could pose a potential security threat if rogue IPv6 routers are present.

Privacy and Security Issues with the EUI-64

EUI-64 is a standard method of deriving the EUI field, the lower 64 bits of the IPv6 address, from the 48 bit MAC address of the associated network interface. Along with some bit manipulations, the 48 bits are expanded to 64 bits by splitting the 48 bits into two 24 bit fields and inserting the 16 bit value fffe between the two halves.

In the EUI-64 case, the EUI field is derived from the network interface MAC address and remains a constant across network address renumbering. There has been some concern over possible privacy issues where a system could be tracked or identified between networks or across renumbering. The IETF addressed this issue by specifying that an address may be identified by a privacy-protecting random EUI chosen in such a way as to never collide with an auto-configured EUI.

In addition to the privacy issue, there exists the possibility of mapping network infrastructure or connectivity through EUI mapping. Mappings of EUIs between networks can reveal underlying subnet structures and subnet mappings (SLAs) between networks.

EUI mappings could also reveal router and serial connectivity linkages, because PPP end-points often inherit EUIs from the system network interface. Sites concerned with privacy issues may wish, by policy, to prohibit the auto-configured EUI-64 End Unit Identifiers to prevent EUI tracking and mapping.

Security Issues with Well Known or Trivial EUI Values

Some configurations of 6to4 default to well-known values for the EUI field. Often, this configuration simply uses the value of 1 or 2 for the EUI field. This process is also frequently true for network administrators setting up routers or well-known services such as DNS, NTP, or virtual hosting addresses on Web servers. In some cases, it may be the IPv4 address of the interface used as an EUI value.

The use of well-known EUI values reduces the scope required to scan for IPv6 addresses. This is especially true in the case of 6to4, where an additional 32 bits (the IPv4 address) are already known and the SLA value typically defaults to a well-known value. Since 6to4 is on prefix 2002::/16 with the IPv4 address in the next 32 bits and guessable values for SLA and EUI, it's no more difficult to scan for systems with a default 6to4 address configured than it is to scan for any IPv4 port.

In the case of Windows XP, the default 6to4 address is 2002 followed by the IPv4 address, followed by 0 for the SLA, followed by 32 bits of zeros for the upper half of the EUI, followed by the IPv4 address, once again, in the lower 32 bits of the EUI. Linux, by default, uses the value of 1 for the EUI in a 6to4 address. This designation makes both assignments totally predictable based on the IPv4 address. It also makes them unique from each other and, thus, possible to identify by 6to4 fingerprinting techniques.

Example:

IPv4: 10.130.205.16
6to4 on XP: 2002:0a82:cd10::0a82:cd10
6to4 on Linux: 2002:0a82:cd10::1

The use of well-known EUI values for routers or well-known services is totally unnecessary in IPv6. This practice should be severely discouraged due to the adverse impact on resistance to network scanning.

Not every site will restrict the use of trivial, simplistic, or predictable EUI values. This usage should be a concern of policy and best practices. Sites concerned with simplistic EUI address values can set a Network IDS to detect IPv6 traffic with local addresses where there are excessive "all zeros" or "all ones" in the high order bytes of the EUI half of the address, bits 64 through 95.

IPv6 Stateless Autoconfiguration

Hosts that look for autoconfiguration routers generate router solicitation and neighbor solicitation requests. These requests should not be present on networks that do not supporting or provide IPv6. Their presence or corresponding responses expose systems that have altered configurations for IPv6 support.

Router advertisement packets indicate the presence of an active IPv6 router on a local subnetwork. If IPv6 support is not sanctioned, this type of traffic usually indicates the presence of a rogue IPv6 router.

Router advertisements, router solicitations, and neighbor solicitations are normal on IPv6 networks. Autoconfiguration makes the creation of rogue gateways on IPv6 extremely simple. Consequently, attention should be paid to any unusual router advertisements. This would include:

- Routers advertising the same established prefixes
- Routers advertising any new prefixes
- Prefix changes outside of renumbering and transition periods

Prefix changes and router changes should be rare. Abnormal router or prefix changes are at least as serious a cause for concern as duplicate IP addresses or rogue routing protocol traffic on IPv4.

SIT

SIT is the standard for tunneling or encapsulating IPv6 over IPv4. SIT is also supported under GIF (General IP Forwarding) in the BSD operating systems. SIT is listed as IP protocol 41 (ipv6) for assigned IPv4 protocols.

If IPv6 is not provided or supported, any form of SIT traffic would be abnormal and indicative of possibly malicious traffic. If IPv6 is being provided and supported, SIT traffic and tunnels to and from infrastructure routers and gateways are normal.

6to4
The 2002::/16 prefix was allocated for use by 6to4 automatic SIT tunnels on IPv4 hosts with no external IPv6 support and no static-configured SIT tunnels. 6To4 can be readily configured on supporting systems and used to establish IPv6 based connections between individual IPv4 hosts with no actual IPv6 network present at either end, or anywhere in between. This special category of SIT tunnels uses the SIT protocol with a special purpose IPv6 prefix to autoconfigure the tunnel endpoints.

For 6to4 to communicate with the other two global prefixes, a gateway is required. Standards have defined certain "anycast" addresses on the core IPv4 Internet to provide gateways between 6to4 and the 6bone and Internet6. All three of the global top-level prefixes interoperate and communicate with each other regardless of differences in allocation schemes, management schemes, and routing.

Because the source gateways do not require static configuration of the endpoints, it is possible to direct 6to4 packets at a destination gateway that does not support IPv6 or SIT. This ability opens up the possibility of DDoS attacks against IPv4 hosts from IPv6 networks even where no IPv6 or SIT support exists on the target systems.

Addresses with 2002 in the high order word of the IPv6 address and with a non-local IPv4 address in the next 32 bits are often normal traffic when communicating with remote 6to4 nodes. They do not necessarily indicate malicious activity.

Protection in an IPv6 World

The risks introduced primarily by IPv6 transition mechanisms can be mitigated and controlled using existing applications and techniques. Unconstrained 6to4 should simply be blocked. 6To4 is a transitional mechanism intended for individual independent nodes to connect IPv6 over the greater Internet. Any large network should provide orchestrated routing for IPv6 where IPv6 is supported, so there should be no need for unconstrained 6to4. Likewise, if IPv6 is not going to be supported, there is no need for SIT/IPv6 tunnels at all.

Fortunately, most good firewalls will block IP protocol 41, the 6to4 and SIT tunnel protocol, unless it has been explicitly enabled. Thus, most firewalls by default block SIT tunnels and 6to4 tunnels. Administrators need to verify and test this setting. This blocking configuration does not protect against Teredo-style IPv6 over UDP tunnels, but these tunnels are relatively rare due to lack of support and lack of mature standards. Teredo is likely to become more common in the near future as the IETF completes work on the standard.

If IPv6 is supported, the SIT tunnels should be provided by the infrastructure system (routers and gateways) whether or not 6to4 auto-tunneling is supported. Delivery of IPv6 traffic to individual nodes and workstations should be done via native IPv6. SIT traffic (static or 6to4 auto tunnel) to and from individual workstations then becomes an immediate indicator of potential trouble.

If 6to4 is supported, it should be supported intentionally from a limited number of well-defined gateways. This procedure provides direct access support from individual 6to4 nodes on the external networks while controlling the traffic flows on the inner, protected network.

Networks should also be monitored for IPv6 auto-configuration packets, router and neighbor discovery, solicitation and advertisement packets. If IPv6 is not supported on a specific network segment, it is an indication of possible misconfiguration, rogue tunnels and routing, or malicious actively. If IPv6 is being supported, router advertisements and solicitations should be monitored for rogue routers, indicative of possible intrusions or backdoors. Any non-infrastructure device advertising a new IPv6 route or prefix should be immediately flagged for investigation.

Firewalls and IPv6

Most firewalls with IPv6 support have separate rule-sets for IPv6 and IPv4. These rule-sets must be coordinated and consistent to be properly managed and to avoid an inadvertent security exposure.

Tunneling protocols such as SIT and Teredo present particular challenges for firewalls. Few, if any, are designed to unroll these tunneling protocols and apply rule sets directly to the encapsulated traffic. Bit mapped rules, which mask against encapsulated payloads, are a difficult and error-prone workaround.

To an IPv4 firewall rule-set, SIT and 6to4 are nothing more than IP protocol 41 on IPv4. To an IPv6 firewall rule-set, SIT and 6to4 do not exist. Neither rule set applies directly to these tunnels beyond switching protocol 41 on or off. In a similar vein, Teredo is nothing more than a UDP protocol on IPv4, and is not seen by the IPv6 stack and rule-set.

These tunneling protocols represent a significant hole through firewalls lacking rules that can be applied directly against the tunneled payload traffic, and should be blocked from forwarding across security perimeters and across firewalls.

To support IPv6, transitional tunnels should be terminated at or outside the network security perimeter and firewalls, and routed natively through the firewall where appropriate rules and tests can be applied.

Internet Security Systems' Dynamic Threat Protection™ IPv6 Solutions

Internet Security Systems' RealSecure® 7.0 Network intrusion protection application and Proventia™ protection appliances support several current IPv6 standards and tunneling mechanisms. RealSecure and Proventia have the ability to decode and inspect all supported IPv4 protocols and any associated attacks or misuse, even if encapsulated in IPv6. RealSecure and Proventia completely decode and inspect IPv6 encapsulated in SIT tunnels; either static SIT tunnels or 6to4 auto SIT tunnels. In short, these Dynamic Threat Protection solutions already provide sophisticated, automated protection against IPv6 threats, including networks not yet configured for IPv6 operations.

Detection of Native IPv6 traffic
On networks without IPv6 support, RealSecure or Proventia can detect IP traffic with a version number 6 in the IP header. If the network does not supporting IPv6, this could be indicative of malicious, or at least, non-supported network traffic and a potential security threat if there are rogue IPv6 routers present.

Detection of SIT Traffic
If IPv6 is not being provided or supported, any form of SIT traffic would be abnormal and indicative of possible malicious traffic. RealSecure or Proventia can easily be configured to detect and report this activity.

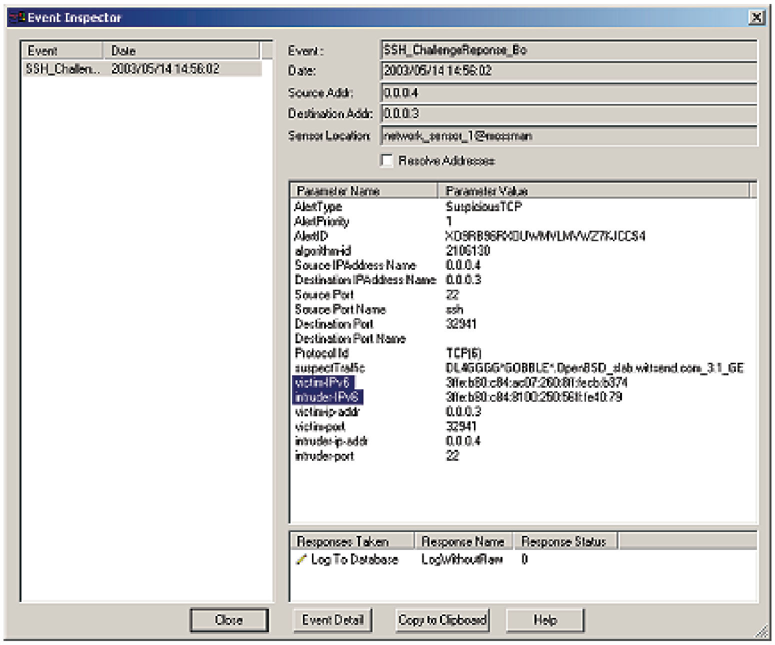
If IPv6 is being provided and supported, SIT traffic and tunnels to and from infrastructure routers and gateways are normal. It should not be normal on IPv6 native links and is indicative of potential rouge gateways and tunnels. RealSecure or Proventia can detect any traffic with an IP protocol value of 41 (IPv6/SIT) or 47 (GRE) where those tunnels are not being specifically supported by the network infrastructure.

Detection of 6to4 Traffic

Whether or not IPv6 support is provided, 6to4 traffic (SIT traffic with a local :2002: prefix) should be abnormal on large managed networks. This presence should be detected and identified by the IPv4 address of the 6to4 gateway. It is strongly indicative of a malicious rogue gateway. Infrastructure devices, such as routers and firewalls providing IPv6 tunnel connectivity, should only be using static SIT tunnels.

RealSecure or Proventia can detect any SIT traffic with either a source or destination address which begins with :2002: and which contains a local IPv4 network address in the next 32 bits of the IPv6 address.

Addresses with :2002: in the high order word of the IPv6 address and with a non-local IPv4 address in the next 32 bits should be normal traffic when communicating with remote 6to4 nodes and do not necessarily indicate malicious activity.



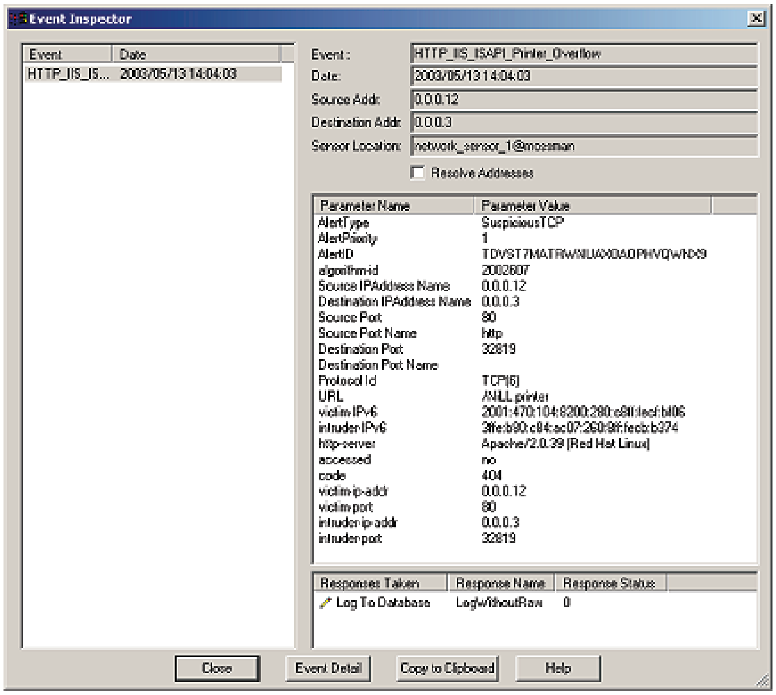
Event	Date
SSH_Challen...	2003/05/14 14:56:02

Event: SSH_ChallengeReponse_Bo
Date: 2003/05/14 14:56:02
Source Addr: 0.0.0.4
Destination Addr: 0.0.0.3
Sensor Location: network_sensor_1@mossman
□ Resolve Addresses

Parameter Name	Parameter Value
AlertType	SuspiciousTCP
AlertPriority	1
AlertID	XD9RB96RXDUWMVLMVWZ7KJCCS4
algorithm-id	2106130
Source IPAddress Name	0.0.0.4
Destination IPAddress Name	0.0.0.3
Source Port	22
Source Port Name	ssh
Destination Port	32941
Destination Port Name	
Protocol Id	TCP(6)
suspectTraffic	DL4GGGG*GOBBLE*.OpenBSD_dab.witsend.com_3.1_GE
victim-IPv6	3ffe:b80:c84:ac07:260:8ff:fecb:b374
intruder-IPv6	3ffe:b80:c84:8100:250:56ff:fe40:79
victim-ip-addr	0.0.0.3
victim-port	32941
intruder-ip-addr	0.0.0.4
intruder-port	22

Responses Taken	Response Name	Response Status
Log To Database	LogWithoutRaw	0

Figure 1. Example RealSecure Network detection event of an IPv4 attack against the OpenSSH service, encapsulated in IPv6 traffic.



Event	Date
HTTP_IIS_IS...	2003/05/13 14:04:03

Event : HTTP_IIS_ISAPI_Printer_Overflow
Date: 2003/05/13 14:04:03
Source Addr: 0.0.0.12
Destination Addr: 0.0.0.3
Sensor Location: network_sensor_1@mossman
☐ Resolve Addresses

Parameter Name	Parameter Value
AlertType	SuspiciousTCP
AlertPriority	1
AlertID	TDVST7MATRWNUAXDAOPHVQWNX9
algorithm-id	2002607
Source IPAddress Name	0.0.0.12
Destination IPAddress Name	0.0.0.3
Source Port	80
Source Port Name	http
Destination Port	32819
Destination Port Name	
Protocol Id	TCP(6)
URL	/NULL.printer
victim-IPv6	2001:470:104:8200:280:c8ff:fecf:bf06
intruder-IPv6	3ffe:b80:c84:ac07:260:8ff:fecb:b374
http-server	Apache/2.0.39 (Red Hat Linux)
accessed	no
code	404
victim-ip-addr	0.0.0.12
victim-port	80
intruder-ip-addr	0.0.0.3
intruder-port	32819

Responses Taken	Response Name	Response Status
Log To Database	LogWithoutRaw	0

Figure 2. Example RealSecure Network detection of the IIS ISAPI Overflow attack used by the Code Red worm. This IPv4 attack was encapsulated in IPv6 and detected.

Conclusion

IPv6 is a new, widely available version of Internet Protocol that carries a number of significant performance and security advantages over earlier versions. However, these same benefits also work to the advantage of IPv6-savvy attackers, since many network administrators have not deployed IPv6, and are unaware that IPv6 traffic can pass through their networks without their awareness.

As with most malicious Internet activity, it is only a matter of time before this elite knowledge filters down to become common knowledge in the Internet underground. IPv4 network administrators are often already behind when it comes to protection against IPv6 abuse. However, strong IPv6 security practices, products and services, such as those provided by Internet Security Systems, are available to protect against these newly emergent threats.

IPv6 offers security advantages to those who can best utilize it. These strengths can be wielded by either the defender or the intruder. Therefore, the time for ignoring IPv6 is past. The time for understanding it, recognizing it and deploying its obvious advantages is now.

References

6Bone information:
http://www.6bone.net

IPv6 tools, protocol bouncers and IPv6 references:
http://www.hs247.com

Relay6 protocol bouncer:
http://francesco.netsigners.com/stpuk/relay6.htm

Underground tools and resources:
http://www.pkcrew.org

Various tools and resources (underground and otherwise):
http://packetstormsecurity.nl

The Honeynet Project:
http://project.honeynet.org

Honeypot Mailinglist Archives:
http://lists.insecure.org/lists/honeypots/

About Internet Security Systems (ISS)

Internet Security Systems, Inc. (ISS) (Nasdaq: ISSX) is a world leader in Dynamic Threat Protection™ products and services that protect critical information assets from an ever-changing spectrum of threats and misuse. Solutions from Internet Security Systems dynamically detect, prevent and respond to sophisticated threats to networks, servers and desktops. Services include 24/7 system monitoring, emergency response and access to the X-Force™, Internet Security Systems' renowned research and development team. Internet Security Systems is the trusted security provider for more than 11,000 corporate customers, including all of the Fortune 50, the top 10 largest U.S. securities firms, 10 of the world's largest telecommunications companies and major agencies and departments within U.S. local, state and federal governments. Headquartered in Atlanta, Ga., Internet Security Systems has additional operations throughout the Americas, Asia, Australia, Europe and the Middle East. For more information, visit the Internet Security Systems Web site at www.iss.net or call 888-901-7477.