International Journal of Advance Research in Science and Engineering
Vol. No.6, Issue No. 01 , January 2017
www.ijarse.com
IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

ARCHITECTURAL SUPPORT FOR SECURITY MANAGEMENT

S.Joseph Gabriel[1], A.Mohammed Afsar Basha[2], P. Rizwan Ahmed[3]

[1]*Associate Professor & Head of Computer Science, Mazharul Uloom College, Ambur, (India)*
[2]*Research Scholar, Mazharul Uloom College, Ambur, (India)*
[3]*Assistant Professor & Head of Computer Application, Mazharul Uloom College, Ambur, (India)*

ABSTRACT

This paper presents a principled approach to network redesign that creates more secure and manageable networks. We propose a new network architecture in which a global security policy defines all connectivity. The policy is declared at a logically centralized Controller and then enforced directly at each switch. All communication must first obtain permission from the Controller before being forwarded by any of the network switches. The Controller manages the policy namespace and performs
all routing and access control decisions, while the switches are reduced to simple forwarding engines that enforce the Controller's decisions.

We present an idealized instantiation of the network architecture called SANE. In SANE, the Controller grants permission to requesting flows by handing out capabilities (encrypted source routes). SANE switches will only forward a packet if it contains a valid capability between the link and network headers. SANE thus introduces a new, low-level protection layer that defines all connectivity on the network. We present the design and prototype implementation, showing that the design can easily scale to networks of tens of thousands of nodes.

***Index Terms*: Packet Filter, Security Management, SANE**

I. INTRODUCTION

Internet architecture was born in a far more innocent era, when there was little need to consider how to defend against malicious attacks. Many of the Internet's primary design goals that were so critical to its success, such as universal connectivity and decentralized control, are now at odds with security.

Worms, malware, and sophisticated attackers mean that security can no longer be ignored. This is particularly true for enterprise networks, where it is unacceptable to lose data, expose private information, or lose system availability. Security measures have been retrofitted to enterprise networks via many mechanisms, including router ACLs, firewalls, NATs, and middleboxes, along with complex link-layer technologies such as VLANs.

Despite years of experience and experimentation, these mechanisms remain far from ideal and have created a management nightmare. Requiring a significant amount of configuration and oversight, they are often limited in the range of policies that they can enforce and produce networks that are complex and brittle. Moreover, even with these techniques, security within the enterprise remains notoriously poor. Worms routinely cause significant losses in productivity [18] and increase potential for data loss. Attacks resulting in theft of

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

intellectual property and other sensitive information are also common. The long and largely unsuccessful struggle to protect enterprise networks convinced us to start over with a clean slate. We aim to design manageable networks with security as a fundamental design property rather than an afterthought. Our approach begins with an idealized network architecture in which we assume that we can replace every networking component (such as switches, routers, and firewalls) as well as end-host networking stacks. Once we arrive at a solution that has the appropriate properties, we address the compatibility issue. We draw from the lessons learned in designing a new architecture from the ground up and apply them to the design and implementation of a more practical solution. This second architecture does not require changes to the end host and can be incrementally deployed within existing networks.

II. PROBLEMS WITH CURRENT APPROACHES

Before describing our approach, we discuss the shortcomings of the architecture most commonly used in today's networks. In particular, we look at the properties that lead to insecurities or those that contribute to the lack of network manageability. Loose Address Bindings and Lack of Attribution Today's networking technologies are largely based on Ethernet and IP, both of which use a destination based datagram model for forwarding. The source address of packets traversing the network are largely ignored by the forwarding elements. This has two important, negative consequences. First, a host can easily forge its source address to evade filtering mechanisms in the network. Source forging is particularly dangerous within a LAN environment where it can be used to poison switch learning tables and ARP caches. Source forging can also be used to fake DNS [15] and DHCP responses. Secondly, lack of in-network knowledge a traffic sources makes it difficult to attribute a packet to a user or to a machine. At its most benign, lack of attribution can make it difficult to track down the location of "phantom-hosts" [13]. More seriously, it may be impossible to determine the source of an intrusion given a sufficiently clever attacker.

III. COMPLEXITY OF MECHANISM

A typical enterprise network today uses several mechanisms simultaneously to protect its network: VLANs, ACLs, firewalls, NATs, and so on. The security policy is distributed among the boxes that implement these mechanisms, making it difficult to correctly implement an enterprise-wide security policy. Configuration is complex; for example, routing protocols often require thousands of lines of policy configuration. Furthermore, the configuration is often dependent on network topology and based on addresses and physical ports, rather than on authenticated end-points. When the topology changes or hosts move, the configuration frequently breaks, requires careful repair [68], and potentially undermines its security policies.

A common response is to put all security policy in one box and at a choke-point in the network, for example, in a firewall at the network's entry and exit points. If an attacker makes it through the firewall, then they will have unfettered access to the whole network. Another way to address this complexity is to enforce protection of the end host via distributed firewalls. While reasonable, this places all trust in the end hosts. End host firewalls can

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

be disabled or bypassed, leaving the network unprotected, and they offer no containment of malicious infrastructure, e.g., a compromised NIDS [17].

IV. SOLUTION REQUIREMENTS

In addition to retaining the characteristics that have resulted in the wide deployment of IP and Ethernet networks – simple use model, suitable (e.g., Gigabit) performance, the ability to scale to support large organizations, and robustness and adaptability to failure – a solution should address the deficiencies addressed in the previous section. In particular, a security management architecture solution should conform to the following central design principles.

1. The network should be managed from a single global security policy declared over mnemonic names. We seek an architecture that supports natural policies that are independent of the topology and the equipment used e.g., "allow everyone in group sales to connect to the http server through a web proxy.". This contrasts with todays policies, which are typically expressed in terms of topology-dependent ACLs in firewalls. Through high-level policies, our goal isto provide access control that is restrictive (i.e. provides least privilege access to resources) yet flexible, so that the network does not become unusable. Through logical centralization, we avoid the distributed maintenance problems that are legion with todays firewalls and ACL configuration files[64].
2. Policy should determine the path that packets follow. There are several reasons for policy to dictate the paths. First, policy might require that packets pass through an intermediate middlebox; for example, a guest user might be required to communicate via a proxy, or the user of an unpatched operating system might be required to communicate via an intrusion detection system [4, 9]. Second traffic can receive more appropriate service if its path is controlled. Directing real-time communications over lightly loaded paths, important communications over redundant paths, and private communications over paths inside a trusted
3. boundary would lead to better service. Allowing the network manager to determine the paths via policy– where the policy is in terms of high-level names– leads to finer-level control and greater visibility than current designs.
4. Minimize the trusted computing base. Today's networks trust multiple components, such as firewalls, switches, routers, DNS, and authentication services (e.g., Kerberos, AD, and Radius). The compromise of any one component can wreak havoc on the entire enterprise. Our goal is to reduce the trusted computing base as much as possible; optimally, the architecture would gracefully manage malicious switches.

Threat Environment

In the proposed architecture, we seek to provide protection robust enough for demanding threat environments, such as government and military networks, yet flexible enough for everyday use. We assume a robust threat environment with both insider (authenticated users or switches) and outsider threats (e.g., an unauthenticated attacker plugging into a network jack). This attacker may be capable of compromising infrastructure

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

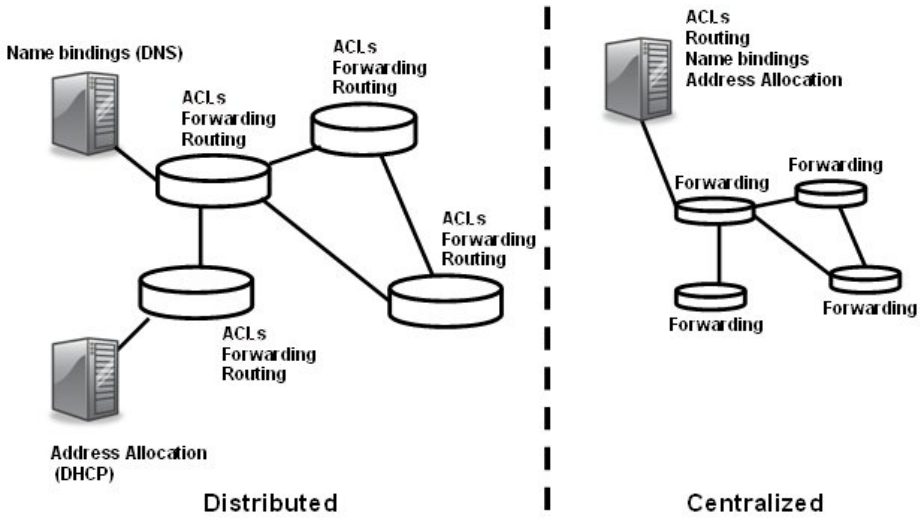
components and exploiting protocol weaknesses; consequently, we assume that attacks can originate from any network element, such as end hosts, switches, or firewalls.

Our goal is to prevent malicious end hosts from sending traffic anywhere that has not been explicitly authorized or, if authorized, subjecting the network to a denial of service attack that cannot be subsequently disabled. Our solution also makes an attempt to maintain availability in the face of malicious switches.

V. A CENTRALIZED, DEFAULT-OFF SOLUTION

In order to achieve the properties described in the previous section, we choose to build our designs around a centralized control architecture. We feel that centralization is the proper approach to build a secure and manageable network for the enterprise. IP's best effort service is both simple and unchanging, which makes it well-suited for distributed algorithms. Network security management is quite the opposite: its requirements are complex and require strong consistency, making it quite difficult to compute in a distributed manner.

Both of the proposed architectures in this thesis are managed from a logically centralized Controller. In our approach, rather than distributing policy declaration, routing computation, and permission checks among the switches and routers, these functions are all managed by the Controller. As a result, the switches are reduced to very simple, forwarding elements whose sole purpose is to enforce the Controllers decisions. Figure 1.1 shows how, in a centralized architecture, a single Controller subsumes functionality handled today by routers and special purpose servers.



Centralizing the control functions provides the following benefits. First, it reduces the trusted computing base by minimizing the number of heavily trusted components on the network to one. This is in contrast today in which a compromise of any of the trusted services, LDAP, DNS, DHCP, or routers can wreak havoc on a network.

Secondly, limiting the consistency protocols between highly trusted entities protects them from attack. Today consistency protocols are often done in plaintext (e.g. dyndns) can thus be subverted by a malicious party with access to the traffic. Finally, centralization reduces the overhead required to maintain consistency.

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

While there are many standard objections to centralized approaches, such as resilience and scalability. As discussed in sections 2.2.5 and 3.3.5 standard replication techniques can provide excellent resilience. Current CPU speeds make it possible to manage all control functions on a sizable network (e.g., 25,000 hosts) from a single commodity PC. Another design choice was to make the network "off-by-default". That is, by default, hosts on the network cannot communicate with each other; they can only route to the network Controller. Hosts and users must first authenticate themselves with the Controller before they can request access to the network resources and, ultimately, to other end hosts. Allowing the Controller to interpose on each communication allows strict control over all network flows. In addition, requiring authentication of all network principles (hosts and users) allows control to be defined over high level names in a secure manner.

At first glance, our approach may seem draconian, as all communication requires the permission of a central administrator. In practice, however, the administrator is free to implement a wide variety of policies that vary from strict to relaxed and differ among users and services. The key is that our approach allows easy implementation and enforcement through centralization of a simply expressed network security policy.

VI. CONCLUSIONS

This paper described a new approach to dealing with the security and management problems found in today's Enterprise networks. Ethernet and IP networks are not well suited to address these demands. Their shortcomings are many fold. First, they do not provide a usable namespace because the name-to-address bindings and address-to principle bindings are loose and insecure. Secondly, policy declaration is normally over low-level identifiers (e.g., IP addresses, VLANs, physical ports and MAC addresses) that don't have clear mappings to network principles and are topology dependant. Encoding topology in policy results in brittle networks whose semantics change with the movement of components. Finally, policy today is declared in many files over multiple components. This requires the human operator to perform the labor intensive and error prone process of manual consistency.

REFERENCES

[1] 802.1D MAC Bridges. http://www.ieee802.org/1/pages/802.1D-2003.html.
[2] Apani home page. http://www.apani.com/.
[3] Berkeleydb. http://www.oracle.com/database/berkeley-db.html.
[4] Cisco network admission control.http://www.cisco.com/en/US/netsol/ns466/networking solutions package.html.
[5] Consentry. http://www.consentry.com/.
[6] DNS Service Discover (DNS-SD). http://www.dns-sd.org/.
[7] Identity engines. http://www.idengines.com/.
[8] Lumeta. http://www.lumeta.com/.
[9] Microsoft network access protection. http://www.microsoft.com/technet/network/nap/default.mspx.
[10] Openwrt. http://openwrt.org/.
[11] Packetmotion home page. http://www.packetmotion.com/.
[12] Securify. http://www.securify.com/.

IJARSE
ISSN (O) 2319 - 8354
ISSN (P) 2319 - 8346

[13] Tracking down the phantom host. http://www.securityfocus.com/infocus/1705.

[14] UPnP Standards. http://www.upnp.org/.

[15] Zodiac: Dns protocol monitoring and spoofing program. www.packetfactory.net/projects/zodiac/.

[16] Cisco Security Advisory: Cisco IOS Remote Router Crash.
http://www.cisco.com/warp/public/770/ioslogin-pub.shtml, August 1998.

[17] CERT Advisory CA-2003-13 Multiple Vulnerabilities in Snort Preprocessors.
http://www.cert.org/advisories/CA-2003-13.html, April 2003.

[18] Sasser Worms Continue to Threaten Corporate Productivity.
http://www.esecurityplanet.com/alerts/article.php/3349321, May 2004.