Volume : 2 | Issue : 7 | July 2013 • ISSN No 2277 - 8179

Research Paper

Security in Multicast DNS

Computer Science

KEYWORDS : Multicast, DNS, DNSSEC, IPv6, IPSEC

Sridevi — Assistant Professor, Department of Computer Science, Karnatak University, Dharwad.

Dr.Manjaiah D.H — Professor, Department of Computer Science, Mangalore University, Managlore

ABSTRACT

This paper looks at how developers can use open and freely available technology to provide a level of identity security and trust to service discovery in an untrusted ad-hoc network environment. Initial research shows that using multicast Domain Name Services (DNS) when coupled with X.509 machine-issued certificates gives an adequate level of identification when new computers wish to find and use services across a network. Further research is necessary to generalize the problem and to explore caveats.

1. Introduction

With the advent of pervasive and ubiquitous computing, society has come to depend on technology as part of daily life. Advances in human factors engineering and pioneering work by high tech companies have made computers extremely easy to use. What the average user does not realize, however, is the amount of complexity that goes into even the simplest of tasks. As a result of design patterns, development techniques, and advances in research, some of those complexities are abstracted out in the form of application programming interfaces (APIs) or lower level processes that leave the developer or the user to focus on less mundane tasks.

In the field of network services, developers are starting to reach the stage of making these services packaged and user friendly. Real-world examples include printers that automatically appear in a shared network, or music libraries that become instantly discoverable and sharable [1]. In both of these examples, no knowledge of configuration parameters is needed to take advantage of them, making it easier for the end user to interact with these components and allowing the user to complete the task with minimal effort and without knowledge of underlying details.

Malicious users are frequently overlooked in these environments. Sadly, cyberspace has grown into an untrusted environment where rogue entities passively and actively try to exploit holes in a network for their own personal gain. Because networks were originally built to share information and not to thwart adversaries' attempts to break the infrastructure, problems occur. Therefore, as technology advances it is imperative that security is not overlooked when planning new protocols, services, or systems.This paper explores ways to secure service discovery, especially in situations where a fluid, ad hoc network topology exists. The security needs to do a good job at making sure the systems a local computer is interacting with are trusted. It must ensure this in such a way that it is not overly obstructive and does not prevent the end user from doing his or her task.

2. Multicast DNS

Multicast DNS is an implementation of a DNS server that listens and responds on a multicast address and port. The behavior of a multicast DNS server should be exactly the same as unicast DNS, in which a client can query any and all DNS servers on the multicast group and, if a match is found, a response is returned on that multicast group. The multicast address is defined to be 224.0.0.251, and the port is 5353. The internet draft proposal for multicast DNS states that any query for a DNS record with the .local suffix must be sent over multicast[1]. Queries and responses in multicast DNS are structured in the same way as unicast DNS[2].The most common multicast DNS server is called the multicast DNS responder, which is known as mDNSResponder on Macintosh systems, as mDNSResponder.exe on Windows, and as mdnsd on Unix systems[2].

To prevent excessive network traffic, multicast DNS typically follows some general best practices. First, responses are sent over multicast so that all nodes on the network become aware of the service. If there is no response to the initial query due to either dropped packets or no multicast DNS servers existing on the network, the DNS client will wait a period of time and then retransmit the query again over multicast. If there are multiple hosts on a network that all match a query, they will all respond. In future queries, a client can specify what is called a Known Answer List, meaning the client already knows about certain hosts and is looking for additional hosts. If a multicast DNS server sees a query and recognizes that it is already on the Known Answer List, it will not respond. This is common when a system is constantly polling the network at a certain interval (perhaps every 10 minutes) for new systems on the network. DNS responses also contain a time to live, which specifies how long the multicast DNS clients can treat the response as being valid in cache. In addition to probing the network through queries, multicast DNS also supports periodic announcements on the multicast group. The announcement is structured in the form of a DNS response, which allows all nodes on the network to update their local cache of information on the network[2].

2.1 Secure DNS (DNSSEC)

DNS Security Extensions (DNSSEC) is "a suite of extensions that add security to the DNS protocol"[3]. DNSSEC allows for a DNS domain to be signed with a key, certifying that the records are authentic and have not been tampered with. A top-level domain certifies the legitimacy of its child domains, which in turn sign for their children. DNSSEC uses new DNS record types such as DNSKEY, RRSIG, NSEC, and DS to support the integrity of its records[4]. DNSSEC is supported in both BIND and Windows Server 2008 R2[3]. Not all top-level domains (such as .com and .net) are currently signed. Furthermore, the root zone is not signed[6]. Without these zones being signed, DNSSEC cannot be fully implemented.

2.1.1 DNSSEC lookaside validation

DNSSEC lookaside validation (DLV) is a short-term solution to providing secure DNS without having a signed root or parent zone[6,7]. Domain lookaside validation works by implementing trusts to other domains in the form of DNS DLV records. For example, domains in the .com top-level domain can trust .org. Top-level domains are not the only domains that can implement DLV records. Any domain can implement a DLV to a top-level domain[7]. Another similar temporary solution is to use trust anchors. Trust anchors are provided by the Internet Assigned Numbers Authority (IANA) for signing top-level domains only. From the trust anchors, the top-level domains can implement a DNSSEC hierarchy. Once the root zone is signed, the IANA trust anchor repository will cease to exist[8].

2.1.2 Security concerns in DNSSEC

With the existence of NXT or NSEC records, it is possible for an outsider to discover the entire structure of the DNS domain, which can expose hidden records (such as contact information, public keys, and directory information). The counter argument,

however, is by using DNSSEC, the information is freely available. The zone is signed so nobody can tamper with it[9]. When comparing DNSSEC to certificates in securing a DNS structure, EKRon believes that DNSSEC is the better technology because internet-based certification authorities tend to be less secure[10].

2.2 PKI certificate DNS records
According to RFC 4398, DNS servers allow for the storage of certificate records, including PKIX, SPKI, and PGP records. These records are commonly known as a CERT resource record. The RFC allows for X.509 certificates that contain the user certificate, the certificate authority certificate, the authority revocation list, and the certificate revocation list. The RFC also allows for purpose-based certificates, such as TLS, S/MIME, and IPSec. However, if the DNS response is too big to fit in a UDP payload (for example, the certificate with all of the trusts contained within it), a URL with a link to the certificate could also be substituted, which is known as an indirect type DNS record[9]. An example of a CERT record is the domain foo.com. In addition to having a CNAME record for www, there can be a CERT record in the domain that contains the SSL certificate for the www.foo.com website. Josefsson advocates the use of DNS as a certificate directory for a domain[11].

2.3Bonjour (DNS-SD) and Zeroconf
DNS Service Discovery is commonly known by its trademarked name of Bonjour. Service discovery is part of the larger initiative of zero configuration networking (zeroconf), a way for computers to configure themselves on a network without user intervention. DNS service discovery leverages these SRV records to find a service on the network. SRV records follow the naming convention of instance. Servicetype. domain, so an example of a printer using the internet printing protocol (IPP) type of service would have a service record of some_printer_name.ipp.tcp.example.com[2]. Since service records behave the same way in both standard unicast DNS and multicast DNS, a SRV record for multicast DNS would look like some_printer_name.ipp.tcp.local. It is important to note that "DNS Service Discovery is compatible with, but not dependent on, Multicast DNS."[12]

2.3.1 Working of Query/response in DNS-SD
Tools exist that allow for easy use of DNS-based service discovery. The most common system is DNS-SD.exe. To make a service discoverable, the service must be registered as a service. This is accomplished by running the register command (example dns-sd -R test _daap._tcp . 1234). At this point, there is a service named test that is available on TCP port 1234. Any system on a network that does a query for the test service on the multicast DNS port will get a response [13].

2.4 Universal Plug and Play
Universal Plug and Play is another service discovery and zeroconf technology. It is based on ISO/IEC 29341. Universal Plug and Play has been around since 1999 and has been implemented in versions of Microsoft Windows as early as Windows ME. Running on TCP/IP and HTTP, Universal Plug and Play implements the Simple Service Discovery Protocol to advertise itself on the network. The description of

the service is written in XML. The XML contains URLs to control the device, manage device events, and pull presentation information from a device [14].

2.5 Distributed registries
Popular with service oriented architectures, online registries exist where a user or system can advertise a service. Nodes on a network will have to become aware of an online central registry and use it every time a service is desired. This is analogous to a phone book that advertises services in specific categories (yellow pages). The burden is on the registry to make sure the services advertised in its registry are legitimate. Trabelsi's work on distributed registry models allows for secure access when storing new or existing records, and anonymous access for queries[15]. Trabelsi also goes on to ensure that these registries are resilient to attackers who wish to simply destroy the registry rather than take it over [16].

2.6 Internet protocol version 6 (IPv6)
IP version 6 (IPv6) is a new protocol for internet addressing. It was designed by the Internet Engineering Task Force as a replacement for the current IPv4 standard of addressing [17]. RFC 2460 provides the specification for the protocol. Instead of using a 32-bit field to address nodes on the internet as in IPv4, IPv6 uses 128 bits, which allows for a larger number of addressable nodes. Furthermore, the IPv6 specification has greater support for multicast by adding a new scope field to the address [18].

2.7 Secure IP (IPSec)
IPSec is a means of securing a network connection. IPSec was approved by the Internet Engineering Task Force to provide a point-to-point secure exchange of IP packets. IPSec can be deployed in transport or tunnel modes [19]. Transport mode provides a secure connection between two nodes by simply encrypting the payload. Tunnel mode encrypts and protects the entire IP packet by encrypting the entire message and wrapping it in a new IP packet. IPSec allows for virtual private networking (VPN) because the inner IP packet can be a private non-routable IP address [20]. IPSec is a mandatory part of IP6[21].

There is also limited support for multicast in IPSec. Given that IPSec is designed for secure point-to-point communication, the notion of trying to facilitate key exchange among all members in the multicast group proves to be a challenge. This is especially the case when each node is both a producer and consumer and may join or leave a multicast group at any given moment. While authentication using IPsec may seem possible, encryption certainly is not. Most articles that are looking at multicast over IPSec are focusing on more traditional virtual private networking (VPN)-like uses of IPSec or Generic Routing Encapsulation GRE tunnels [22].

3 Findings
Proposal system suggested that DNSSEC might be able to solve the problem of securing multicast DNS. However, DNSSEC is inadequate and that DNS CERT records would be a better solution. Universal Plug and Play was not considered as a solution because it was too difficult to understand the API. That does not mean Universal Plug and Play is not a viable solution, but further research is necessary.

3.1 Scope of problem
The scope of the problem is limited to a local area network without a gateway to an external network. Time constraints limited the options space that could be explored. The following assumptions are made simplify the problem:

- The network is running IPv4 only, without IPSec installed
- Each node on the network has a unique IP address
- Each machine has a unique PKI certificate issued to it by a certification authority
- All machines on the network trust the issuing root certification authority
- There is no central server on the network that is running a standard DNS server or acting as an authentication service
- Nodes do not have a pre-existing direct trust between each other
- No machines have been compromised by a hacker, virus, or other malicious activity

3.2 Reasons for DNSSEC will not work
The first discovery in our research is that DNSSEC alone will not solve the problem because it requires a hierarchical trust structure (for foo.example.com to be secure, example.com must be secure). For the scenario to work, the .local domain will need to be secured first, but this cannot happen because there is no central DNS repository for .local[2]. Also, the computers cannot implement domain lookaside validation, because to do so two computers would have to trust each other.

3.3 CERT records as a solution
An alternative to DNSSEC would be to include CERT records in the multicast DNS server. For example, suppose a computer que-

ries for a service and gets a response from another computer. The node doing the querying can then request a CERT record from the host with the service. If the host responds with a certificate, the querying computer can then validate the certificate against its repository of trusted root certification authorities. This alternative is backwards compatible with existing infrastructure - if a legacy system queries for a service, it will still get a multicast DNS response, but it won't ask for a CERT record. Likewise, if a newer system queries for a service on a legacy system, it will get a service response but no CERT record. The burden is then on the querying node to determine if the legacy system is trustworthy.

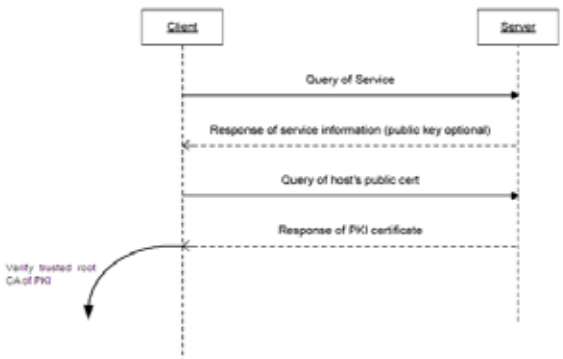

Figure 1: Client/server interaction in service discovery with certificates

3.3.1 Trusted root certification authorities

PKI certificates come in many flavors. Certificates can be issued from a known and reputable source, such as VeriSign, or they can be issued locally by the computer. These reputable sources are known as Certification Authorities (CAs). For the CERT record solution to work, the certificates used by all of the machines in the service discovery need to have a common root CA, or at least know about (and trust) the other CAs. Furthermore, the certification authority does not need to be a globally known commercial entity. If someone is looking to only share a service among a close group of machines in which one person/company has direct control over all systems, then standing up a local CA and issuing certificates to all nodes will be sufficient. Keep in mind, however, that external nodes will not know about the CA and will complain about the validity of the certificates.

3.3.2 User certificates vs. machine certificates

Certificates can be issued to an individual user or to a local machine. There are tradeoffs to using either in the realm of service discovery. In the machine model, it is very easy to query for a host's DNS CERT record by simply knowing the hostname or finding it out via a reverse IP lookup. Trying to find the correct user's certificate in the DNS query is difficult because it is not clear in service discovery which user owns the service. It is important that all services that run on the host use the same certificate. If not, there will be a mismatch between the DNS lookup of the certificate and the service itself. In the user model, the user is the one starting up the service or trying to connect to the service. Therefore, it makes sense to verify that the user is authorized to access the service since many people can use the same computer. The conclusion is that the user's need for access should be determined by a secondary authentication model after the initial service discovery. Two examples are a simple logon or key exchange.

3.4 Support in current technology

In looking at currently available systems, we found that neither BIND, an open source DNS server, nor Windows Server appear to support CERT records. It is also unclear if multicast DNS has CERT support. According to Cheshire's DNS-based service discovery document (on which Bonjour is based), it does not appear that support exists there either. The only supported types used in multicast DNS are A, PTR, TXT, and SRV records [23]. It is possible in theory to configure a traditional DNS server to listen on a multicast address, although there is no evidence of that being encouraged or supported. An alternative is to develop a new multicast DNS responder application with that support natively available, since multicast API libraries do exist [24].

Figure 2: Windows XP does not support CERT records in nslookup

4 Conclusions

In conclusion, it is feasible to secure service discovery through multicast DNS. The findings represent a possible solution to the problem area of trusted service discovery. However, it is by no means the one and only solution. The DNS certificate solution proposed will not make the system ultimately secure. Instead it is merely one piece in the complex security puzzle. One example not covered in this document is ensuring that the computer itself is not compromised. Even if the client knows that a computer offering up services was at one point trusted, there is no guarantee it will remain secure.

REFERENCE

[1] Ceshire, Stuart and Krochmal, Marc. Multicast DNS. multicastdns.org. Apple Inc., September 10, 2008. IETF Internet Draft. http://files.multicastdns.org/draft-cheshire-dnsext-multicastdns.txt. | [2] Cheshire, Stuart and Steinberg, Daniel H. Zero Configuration Networking: The Definitive Guide. 1st Edition. Sebastopol : O'Reilly Books, 2006. IBSN 0-596-10100-7. | [3]Microsoft Corporation. Domain Name System Security Extensions. Microsoft. [Online] 2009. http://www.microsoft.com/downloads/details.aspx?FamilyID=7A005A14-F740-4689-8C43-9952B5C3D36F&displaylang=en. | [4] Clegg, Alan. DNSSEC in 6 minutes. Internet Systems Consortium. [Online] 1.5, 2008. [Cited: September 20, 2009.] Powerpoint presentation in PDF form. https://www.isc.org/files/ | DNSSEC_in_6_minutes.pdf. | [5] Albitz, Paul and Liu, Cricket. DNS and Bind. 4th Edition. Sebastopol : O'Reilly Books, 2001. IBSN 0-596-00158-4. | [6] Afilias. ISC, Afilias and Neustar Bring Secure DNS One Step Closer. Afilias. Afilias, August 10, 2009. http://www.afilias.info/news/2009/08/05/isc-afilias-and-neustar-bring-secure-dns-one-stepcloser. | [7] Weiler, S. DNSSEC Lookaside Validation (DLV). DNSSEC Lookaside Validation (DLV). The Internet Engineering Task Force (IETF), November 2007. http://tools.ietf.org/html/rfc5074. RFC 5074. | [8] IANA. Interim Trust Anchor Repository. Internet Assigned Numbers Authority. Internet Corporation for Assigned Names and Numbers. https://itar.iana.org/. | [9] Josefsson, S. Storing Certificates in the Domain Name System (DNS). IETF March 2006. http://tools.ietf.org/html/rfc4398. RFC 4398. | [10] EKRon. DNSSEC versus certificates. Educated Guesswork. [Online] January 18, 2009. http://www.educatedguesswork.org/2009/01/dnssec_versus_certificates.html. | [11] Josefsson, Simon. LDAP and DNS as Certificate Directories. Simon Josefsson's | Master's Thesis Simon Josefsson, January 7, 2002. http://josefsson.org/master-thesis/node9.html. | [12] Ceshire, Stuart. DNS Service Discovery (DNS-SD) 2009. http://www.dns-sd.org. | 13. DNS Based Service Discovery (DNSSD). Smallegan. June 9, 2005. | http://www.smallegan.com/blog/images/dnssd/ss4.jpg. | [14] Microsoft Corporation. Understanding Universal Plug and Play. UPnP Forum.June 2000. http://www.upnp.org/download/UPNP_UnderstandingUPNP.doc. | [15] Secure Service Discover with Distributed Registries. Trabelsi, Slim and Roudier, Yves. s.l. : IEEE, 2008, pp. 1-6. 978-1-4244-3062-8. | [16] On the Impact of DoS Attacks on Secure Service Discovery. Trabelsi, Slim, Guillaume, Urvoy-Keller and Roudier, Yves. s.l. : IEEE, 2008, pp. 532-537. 978-7695-3492-3. | [17] IPv6: The Next Generation Internet. [Online] IPv6 Information Page, April 14, 2003. http://www.ipv6.org/. | [18] Deering, S and Hinden, R. Internet Protocol, Version 6 (IPv6) Specification. USC | Information Sciences Institute. Network Working Group, December 1998. ftp://ftp.isi.edu/in-notes/rfc2460.txt. RFC 2460. | [19] Webopedia. What is IPSec? Webopedia. [Online] WebMediaBrands Inc., May 20, 2004. http://www.webopedia.com/TERM/I/IPsec.html. | [20] Friedl, Steve. An Illustrated Guide to IPsec. Unixwiz.net - Software Consulting Central. August 24, 2005. http://unixwiz.net/techtips/iguide-ipsec.html. | [21] Das, Kaushik. IPSec & IPv6 - Securing the NextGen Internet. IPv6. IPv6 Inc., 2008. http://www.ipv6.com/articles/security/IPsec.htm. | [22] McKeag, Louise. Adding multicast to IPSec. TechWorld. April 20, 2004. http://howto.techworld.com /security/511/adding-multicast-toipsec/. | [23] Cheshire, Stuart and Krochmal, Marc. DNS-Based Service Discovery. DNS-SD Apple Inc., March 10, 2009. http://files.dnssd. | [24] Jepri and Jer. Net::MDNS::Server - Perl extension for a multicast DNS server. Comprehensive Perl Archive Network. Finnish University and Research Network, 2003.http://kobesearch.cpan.org/htdocs/Net-MDNSServer/Net/MDNS/Server.html