

The Need for Client Side Certificates

Russ Davis

Currently, it is estimated that millions of adult Americans have been victims of identity theft [California 03]. Most of the press releases focus on the high profile cases such as the 30,000 people victimized by insiders [CNN 03]. Estimates are that 700,000 people per year in the United States will experience identity theft [ITRC 03]. [Leyden 03] references an FTC survey and indicates there are approximately 10 million American victims of identity theft each year. Most of the discussion centers along the lines of common attack approaches, such as those described in the FDIC Consumer News [FDIC 03].

Most of the high profile cases focus on the monetary gain associated with identity theft. However, we must observe that we are fighting a war on terrorism and there are a number of state sponsored efforts that could be directed against the United States. Consequently, laws, such as the *Identity Theft and Assumption Deterrence Act* [ITADA 98], will not deter the bad actors discussed in this paper. So how could a person in another country launch a crippling attack against US assets?

Consider the following scenario. Personal identity information is collected on US citizens over an extended period of time. An average user would not know that the information has been compromised, as there are no credit flags or other typical indicators. Indeed, you and I may have already been compromised and not yet know it. Now, having amassed large volumes of personal information, simultaneously launch a massive fraud. How many simultaneous victims would be required to disrupt the financial health of the US? Could we absorb 100,000, 1 million, or 10 million simultaneous attacks? What would be the overall ramifications?

So where are we vulnerable to data collection techniques? For the remainder of this paper, I will focus on man-in-the-middle attacks that could be perpetrated by external sources.

Man-In-The-Middle Attacks

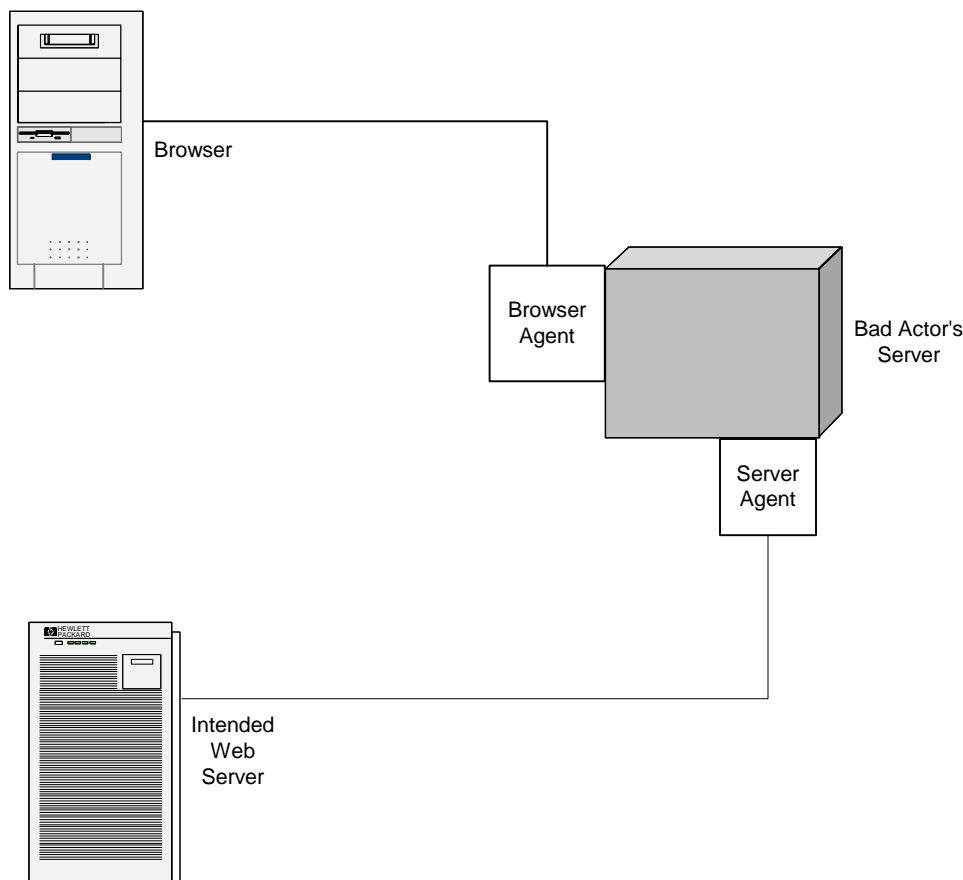
Most of the encryption approaches used over the internet rely on server side authenticated SSL. In this mode, there is only one certificate, on the server, used in establishing the encrypted session. Many of these approaches then require the user to enter in their password as the means to authenticate them. The challenge for the bad actor then is to get the user's ID and Password. SSL protected web vulnerabilities are further discussed in [Burkholder 02] and [Giovanni 99]. So how does a bad actor get in the middle in the first place?

There are numerous papers, source code, tools and the like readily available to bad actors to use. A few examples include [SecuriTeam 91], [TrojanForge 03], [L0t3k 03], and [SANS 01]. Note, there are plenty of references to help any would be bad actor. Once the man-in-the middle attack has started, the bad actor would proxy the information. In this process, valuable information would be collected for use at a later

designated time. Can you state definitively that such information has not already been collected on you?

The following figure depicts a typical man-in-the-middle attack. Here a typical attack, such as a DNS spoof results in the Browser connecting to a bad actor's site instead of the intended web server. First, the user tunnels in to the Browser Agent using SSL (running on the default port 443). The actual connection could either 1) be from the Browser to the bogus server or 2) port 80 (the user could detect this by observing that the lock was open).

Next the Bad Actor's Server Agent connects to the intended web server again using SSL. In effect, there are now 2 SSL connections between the Browser and the intended Web Server. Each screen and response is passed onto the next SSL connection.



Now consider the password authentication. The Intended Web server asks for a password. This request screen is forwarded to the Browser. The user enters the password, this is then forwarded to the Web Server and the bad actor is in.

Now consider a cryptographic challenge and response authentication mechanisms (these are calculator type devices where information is passed from the intended server to

the browser. The user then enters the information into the calculator like device and receives the appropriate response.)

1. The User connection request is forwarded to the intended Web Server.
2. The Web server responds with a cryptographic challenge.
3. The challenge is forwarded from the bad actor's server, to the Web Browser.
4. The user received the cryptographic challenge, enters this value into the calculator device, and responds to the challenge.
5. The Bad actor's server forwards the response to the Web server where the authentication is validated.
6. At this time, there is a secure session with a man-in-the-middle mediating all information flow.

Now lets consider the Secure ID.

1. The User logs into the Bad Actor's server (thinking it the intended Web Server).
2. The Bad Actor's server in turn logs into the actual Intended Web Server.
3. The Intended Web Server requests the user enter the information from the Secure ID token.
4. This request is forwarded back to the Web Browser.
5. The user enters the current information from their Secure ID token.
6. This value is in turn forwarded from the Bad Actor's Web Server to the intended Web Server.
7. The Web Server authenticates the user.
8. At this time there is a secure session with a man-in-the-middle mediating all information flow

Now lets consider a client side certificate.

1. The user logs into the Bad actor's server.
2. The bad actor's server then attempts to establish an SSL session with the intended Web Server.
3. The intended Web Server requests valid client side authentication.
4. As the Bad Actor's web page does not have a valid key, the second SSL connection cannot be established.

So what could a man-in-the middle do to a current session? The ideal goal would be to insert a back door program. If the user stores Word documents on a server that is an avenue for back door exploitation. Alternatively, the Bad Actor may keep a session open long after the Browser thinks the session has concluded.

Conclusion

Given that the FDIC has an Extranet Certification Authority (CA) and can issue certificates, we should. The client browsers readily accept certificates. To ensure the privacy and security of sensitive information within the FDIC's custody, we should take the initiative and provide the necessary security for our customer base.

References:

- [Burkholder 02] *SSL Man-in-the-Middle Attacks*, Peter Burkholder,
<http://www.sans.org/rr/papers/60/480.pdf>
- [California 03] Identity Theft, California Department of Consumer Affairs,
<http://www.privacy.ca.gov/identitytheft.htm>
- [CNN 03] *Tackling Identity Theft*, CNN Headline News
<http://edition.cnn.com/2002/TECH/11/26/hln.wired.id.theft/>
- [FDIC 03] FDIC Consumer News,
<http://www.fdic.gov/consumers/consumer/news/csum00/idthft.html>
- [Giovanni 99] *Bypassing Secure Web Transactions via DNS Corruptions*, Endeavor Systems, <http://downloads.securityfocus.com/library/MiddleMan.pdf>
- [ITADA 98] *Identity Theft and Assumption Deterrence Act*, Public Law 105-318
<http://www.ftc.gov/os/statutes/itada/itadact.htm>
- [ITRC 03] Identity Theft Resource Center, <http://www.idtheftcenter.org/facts.shtml>
- [Leyden 03] *Identity Theft hit 10m Americans a Year*.
<http://www.theregister.co.uk/content/55/32688.html>
- [L0t3k 03] <http://www.l0t3k.org/security/documents/arp/>
- [SANS 01] *Address Resolution Protocol Spoofing and Man-in-the-middle Attacks*, Robert Wagner, <http://www.sans.org/rr/paper.php?id=474>
- [SecuriTeam 91] *Weak authentication in ATT VNC allows man-in-the-middle attack*,
<http://www.securiteam.com/securitynews/5ZPOP1535W.html>
- [TrojanForge 03] *VNC Man in the Middle Exploit Code*,
<http://www.trojanforge.net/showthread/t-5519.html>