

## eSubnet Fragment Article

### A Web of Trust

We see them every day as we surf the public Internet. This little lock icon shows up in different positions depending on the browser used, to provide you with assurance that the site you are visiting is secure.

And this is almost true - but what the lock is actually showing you is that the communication channel between you and the site is secure.

First let's examine what the little lock represents. The lock indicates that the SSL/TLS certificate has the following qualities:

1. The certificate comes from a server whose name matches the name listed in the certificate
2. The time and date range on the certificate are accurate
3. The issuer of the certificate is trusted

If any one of these three qualities is not true, then you will be presented with a certificate error. Today we are looking at the final quality in order to provide a better understanding of the web of trust.

First let us look at a certificate. You can examine one of your own, try your online banking site. Just go to a secured website and double click on the little lock.

On the certificate, you will see three pieces of information.

1. Issued to:
2. Issued by:
3. Valid from \_\_\_\_\_ to \_\_\_\_\_.

These match the qualities mentioned above. If you are looking at a certificate, select the Details tab to see what else you get. Another important item of note is the button 'Install Certificate...', we will return to this later on. **How it all works**

When you go to a site which has a certificate bound to it, the following happens:

1. The client, typically a web browser, sends to the server the encryption type, and some random characters for the encryption process.
2. The web server sends the certificate and its own random string of characters to be used for the encryption process. Included here is the servers 'public key' (The public would have been found under the Details tab if you looked)
3. Upon receipt of the certificate your workstation examines the 3 criteria -
  - a. There is a check to see that the requested URL matches the "Issued To"
  - b. There is a check to see that the date range is valid
  - c. There is a check to see that the "issued by" is trusted ("issued by" is used as insurance against a 'man in the middle attack')
4. If all of these checks pass then a the client generates a premaster secret which it will use for the rest of the session, this is sent off to the server along with the servers 'public key' thus creating a "master secret"
5. With the master secret in place both the browser and server not only communicate through a unique encrypted tunnel, but they can verify that the data sent over the tunnel can not be tampered with.

#### Okay, now what?

Those of you who have purchased certificates (typically from Verisign, Thawte or Entrust) have seen the kind of pricing associated with them. The cost of certificates may have deterred you in the past from properly securing sensitive data on your network, or more importantly with your Intra-net environment.

We now return back to that "Install certificate" button. You can be your own Certificate Authority (CA) just like Verisign and the rest of them. But, there is a catch. Your computer comes with a series of preinstalled certificates. These are the 'root' certificates and come from Verisign, Thawte, Entrust and others. Your workstation does not know about the 'root' certificate from your own CA.

Please read the documentation for the server you are running about setting it up as a CA; you then issue the certificate request, as you would if you were paying Verisign and you are almost done. Distribute and install the 'root' certificate from your CA to the clients and there you go, free secure communications. If you are working with computers out side your Organization, you may wish to purchase a certificate from one of the providers, there are options and I highly recommend you look into them. There are security requirements vs. economies of scale issues here. After all, we are all watching the bottom line.

#### More Information...

Man-in-the-middle attacks occur when the communication is intercepted en route. This is where an unauthorized program sends its own certificate back to the client and makes a client request to the server. Rather than the client and server validating with each other, they validate with the unauthorized program.

Originally published May 2008

Republished: June 2008 in [TLOMA](#) Today

**About the Author**

Richard Danielli is the founder and President of eSubnet Enterprises. He has broad expertise in the fields of networking and data security. Based in Toronto, eSubnet provides superior customized solutions for networking and data security. Additional articles are available at <http://www.esubnet.com/fragment.html>.

**Reprints**

1. Commercial use: Commercial newspapers, magazines and websites that wish to reprint an article should contact eSubnet for permission.
2. Non-Commercial use: Any reproduction of content (excerpt or full text) by nonprofit organizations or companies wishing to share information with their members or clients requires that you notify us, whether for print or online usage.

Regarding reprints, contact Mr. Danielli at [rdanielli\(at\)esubnet.com](mailto:rdanielli@esubnet.com) or at (416) 203-1223.