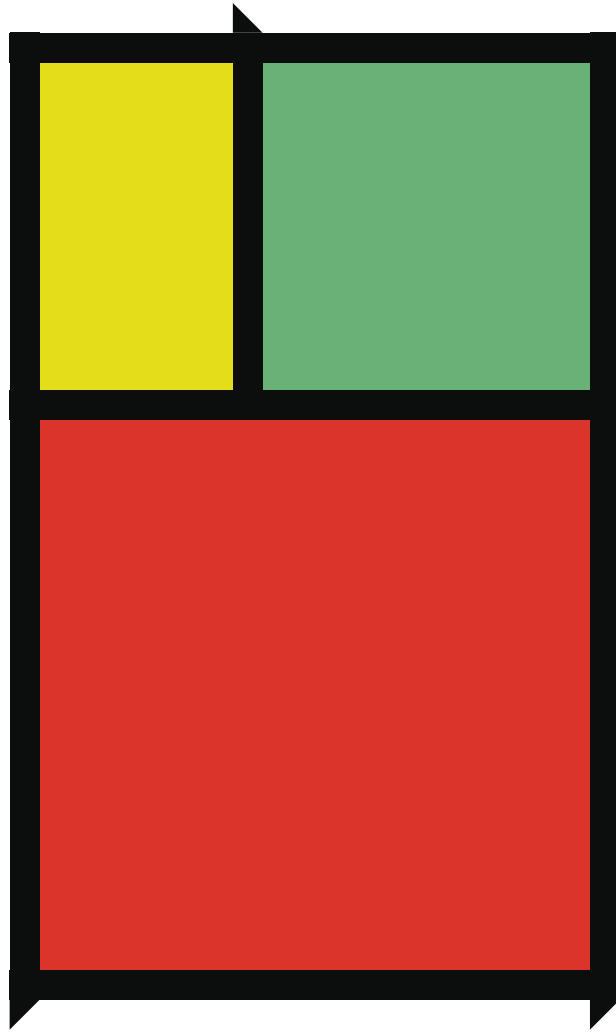


In Partnership with
Green Hills Software



Internet Security for Embedded Systems

Internet Is Not Safe

Since the traditional Internet Protocols have no built-in security features, embedded systems have been avoiding using the Internet. Modern security protocols solve this problem, providing perfectly secure communication over non-secure networks.

The advantages of having the embedded systems connected to the Internet would be tremendous. Today, servicing and software updates almost always require a repairman traveling to the site to make replacements or updates. If the device was connected to the Internet, the operator could make any necessary adjustments or updates from a remote location.

These advantages have been obvious to the world industry for some years already, yet Internet usage is not widely used. The main reason for this is lack

of security. Many embedded systems perform safety-critical missions such as aircraft or factory control; others are business-critical applications within the data and telecom sector. If hackers or competitors gain access, the consequences could be severe.

With the introduction of modern security protocols such as SSH, SSL, IPSec and IKE, the technology is now readily available for secure communication over insecure networks. The many advantages of the Internet are now ready to be explored by the embedded world.

The Internet Protocol Lacks Security

The Internet protocol, IP, has no security features. This means that all data transferred over the protocol is unprotected, which in turn has severe consequences and drawbacks.

Unverified identity of sender and receiver

When communicating over the IP protocol, the identity of the peer is unverified. The IP address is of course known, but unfortunately the address is easy to fake. Thus the IP address provides no reliable information. This is a problem both for clients and servers. Servers typically want to know the identity of the client, e.g. to verify that the client is authorized. Clients typically want to know that they are connected to the correct server, before they start to transfer sensitive information.

Data can be read by unauthorized persons

Data is transferred in clear over the Internet protocol. This means that the data is unprotected from unauthorized reading. Passwords can be stolen, sensitive application data can be recorded as it is transferred, etc.

Data can be modified

There is no way for a receiver to detect if data has been changed since it was sent. If incorrect information is introduced into a system, the consequences are grave. Modified messages also constitute a serious denial-of-service threat.



Non-secure Internet Protocols

Messages can be replayed

An attacker does not have to understand the details of a message. A message can simply be recorded and later replayed. An attacker can record reconfiguration commands to a system. At a later time the messages can be replayed. At that time the configuration commands are probably obsolete, and severe damage could be inflicted to the system. This scenario is of a sabotage or denial-of-service kind, but although these attacks provide no benefit for the attacker, they are an everyday reality on the Internet.

Security Has Been Left to the Application Protocol

Since the IP protocol lacks security features, the responsibility for security has been left to the application protocol. Each application protocol has been forced to provide its own security features. Designing good security requires a significant amount of work, too much work to implement it in each individual protocol. This has led to a situation

where the application protocols only contain few and poor security functions. Some protocols completely lack security!

Telnet – remote terminal protocol

A password is often required when logging in using the telnet protocol. This password is then sent in plain text over the network, exposing it to anyone listening in on the traffic.

FTP – File Transfer Protocol

FTP connections are in plain text. Userid and password sent in order to log in can be stolen by anyone listening in on the communication. Furthermore, the files exchanged are also completely unprotected from unauthorized reading and modification.

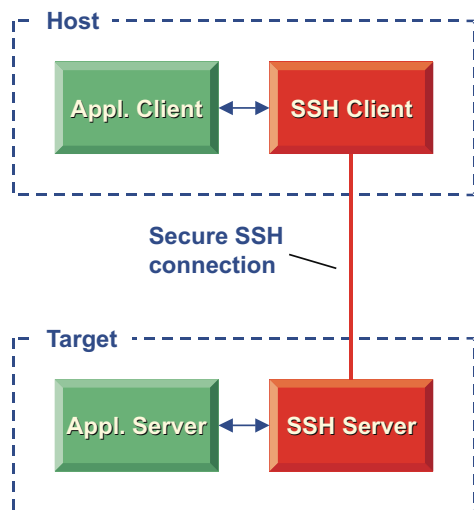
SNMP – Simple Network Management Protocol

The SNMP protocol is used to manage remote equipment. It only uses a community string for authentication, which

is equivalent to only requiring a userid but no password. It is thus very easy to send a faked SNMP request. There is no guarantee that the request has not been changed when transferred over the network. A re-configuration request that has been modified could have devastating effects! Furthermore, replayed requests could also have severe consequences. The lack of security features has severely hindered the deployment of SNMP as a management protocol.

HTTP – Hyper Text Transfer Protocol

The HTTP protocol contains a feature called basic authentication to carry userid and password information. The userid and password is only base64 encoded. It is a trivial task to decode the information and extract the userid and password. In recent years many embedded systems have added a web interface for management. The benefits are obvious; it gives the system a very cheap yet powerful user interface. The security implications are, however, severe. The passwords used to log in can be sniffed, sensitive configuration information is exposed to unauthorized reading, there is no guarantee that configuration updates have not been changed as it was sent over the internet and there is no way to detect replayed messages. telecom and Internet sectors. He is presently a Senior Architect at Interpeak AB (www.interpeak.se). Interpeak is developing state-of-the-art security technology for embedded systems and has numerous security products and complete turn-key embedded security packages.



SSL Security Protocol

SSH, SSL, IPsec and IKE

Secure Shell (SSH), Secure Socket Layer (SSL, also known as Transport Layer Security, TLS), IP Security (IPsec) and Internet Key Exchange (IKE) are modern security protocols. They provide excellent security; authentication, encryption, integrity and replay protection. Thus protection is provided against all the threats described above. SSH, SSL and IPsec/IKE provide very similar security services.

The main difference between the protocols is that the security is applied at different levels in the system structure. SSL resides within the application, while IPsec resides within the TCP/IP stack. SSH is an application of its own, providing a secure connection to a remote host. The protocols are also designed to have a minimal impact on the

application. IPsec is completely transparent to the application, while SSL only requires a very small change to the application. Because the security services are applied on a low level, the application is relieved of security aspects.

All data that arrives to the application is guaranteed to be authentic, not read by unauthorized entities, not modified and not replayed. This makes the application logic significantly easier to design. Furthermore, SSL, IPsec and IKE make it possible to use insecure protocols, in a secure way. SNMP is an excellent example of a protocol that has been difficult to deploy due to lack of security. With the advent of IPsec, SNMP can now be securely used over insecure networks, without changing a single line of code in the SNMP implementation! Since the Internet Engineer-

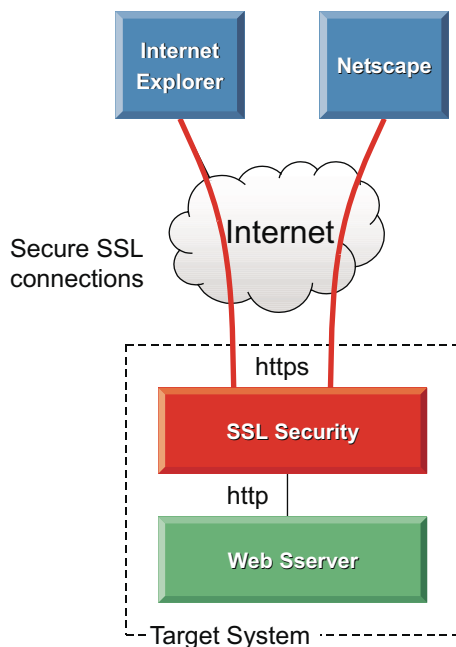
ing Task Force (IETF) specifies both SSL and IPsec, secure interoperability is guaranteed.

Description of SSL

SSL resides between the TCP/IP stack and the application. When an SSL connection is set up a handshake procedure is first performed. During the handshake, authentication is performed and encryption keys are exchanged in a secure way. The algorithms to use for encryption, integrity and replay detection are also negotiated.

All the information negotiated during the handshake is stored in an SSL session. SSL sessions can be re-used for later connections, which will allow subsequent connections to be established with minimal overhead. This is a big advantage, particularly in real-time embedded systems where a full handshake may take too long in certain situations. The application can establish an SSL connection during start-up, in order to cache an SSL session. When "real" communication is needed, the connection establishment is very swift.

Once an SSL connection is established, all data is encrypted on the sending side and decrypted on the receiving side. Furthermore, the sender provides information that makes it possible for the receiver to verify that the message is from the expected sender, that it has not been modified and is not replayed. All this is transparent to the application logic. SSL is the de-facto standard for a number of application protocols. HTTP is the most well-known, but LDAP, SMTP, IMAP and POP3 are other examples.



IPSec and IKE Protocols

Description of IPSec and IKE

IPSec applies authentication, encryption, integrity checks and replay detection at the network layer. This means that each IP packet can be protected and that the security services are completely transparent to the application. In order to carry out the security services, IPSec needs a number of keys. These keys are securely exchanged by IKE.

IPSec is highly configurable. It is possible to specify what packets security shall be applied to, based on source and destination address and port. This makes it possible to secure all communication or only certain applications to certain destinations. The configuration can be tailored to the specific needs in any situation.

A very good characteristic of IPSec/IKE is that it can secure old applications, without having to modify the application. The fact that IPSec resides in the TCP/IP stack makes this possible. The application communicates using TCP and/or UDP and sees nothing of the security applied. With the

configuration options mentioned above, security can easily be turned on and off at runtime as required. IKE can be set up to exchange keys at start-up. The necessary keys will then be available to IPSec, which can then apply security without latency when applications start to communicate. This is a very good characteristic in real time and embedded systems, and it is very similar to the SSL session handling described above.

IPSec and IKE is gaining acceptance rapidly in the industry. It is clearly the emerging standard for IP security. A large number of well known server and desktop operating systems today support IPSec and IKE, e.g. Solaris, Windows 2000, HP-UX, Linux, etc. 4

Conclusion

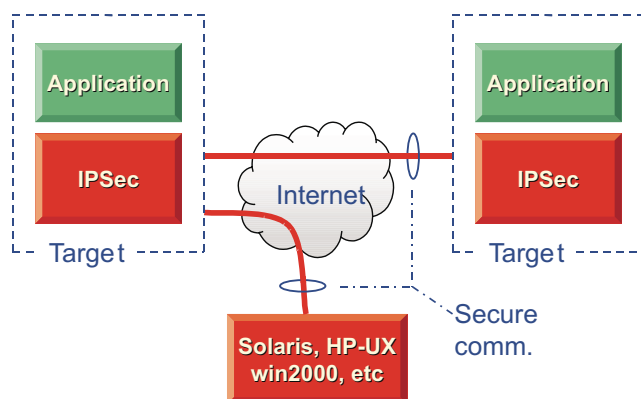
The benefits with having embedded devices connected to the Internet are tremendous, but this has until recently been prevented by security limitations. With the advent of the SSL, IPSec and IKE protocols, it is now possible to securely connect embedded systems.

What protocol to use is mostly governed by the interoperability aspects. If the application protocol is commonly used with SSL such as HTTP, LDAP, etc., SSL is the obvious choice. If IPSec is mostly used or of the application protocol has no established standard for secure communication, IPSec and IKE is the natural choice. The SSL, IPSec and IKE protocols reside below the application, which minimizes or completely eliminates the design impact. The fact that the protocols are specified by IETF ensures secure interoperability.

Security in embedded systems is a business enabler. It lowers costs and creates new possibilities in terms of services and applications. It is now possible to securely connect old insecure applications to the Internet. One area where this will be very common is remote management. One example is that it is now possible to use the SNMP protocol securely over the Internet with IPSec. We will also see a wide range of new services and applications in embedded systems that take advantage of the secure connections that the security protocols provide.

Author

Mr Roger Bodén, M.Sc Mr Bodén has worked with communication security design for eight years within the telecom and Internet sectors. He is presently a Senior Architect at Interpeak AB (www.interpeak.se). Interpeak is developing state-of-the-art security technology for embedded systems and has numerous security products and complete turn-key embedded security packages.



Interpeak Network Security

Interpeak AB, located in Stockholm, Sweden, specializes in network security software and new Internet communication protocols for embedded systems. Interpeak products include IPSec, IKE, SSH, SSL, Web Server Security and NAT. Internet protocols such as LDAP, L2TP, RADIUS, and PPPoE, as well as a dual-mode IPv4/IPv6 TCP/IP stack is also available. For additional information, please visit our homepage: www.interpeak.se, or send a mail to info@interpeak.se.

All Interpeak products are trademarks or registered trademarks of Interpeak AB. Other brand and product names are trademarks or registered trademarks of their respective holders. The information in this document has been carefully reviewed, and is believed to be accurate and reliable. However, Interpeak AB assumes no liabilities for inaccuracies in this document. Furthermore, Interpeak AB reserves the right to change specifications embodied in this document without prior notice.

Version 1.20. Copyright © 2003, Interpeak AB. All rights reserved.