Georgia SoftWorks

Security: SSH Security Risks 🚩 RED FLAG WARNINGS 🚩

Georgia SoftWorks
17 Hwy 9 South
PO Box 567
Dawsonville GA 30534
USA

Proprietary Encryption Protocols:

Question: When should I use Proprietary Encryption Protocols?

Answer: Almost Never.   Your application should be designed in such a way that a standard cryptographic protocol can be used.

Don't be fooled by companies intermixing words like *proprietary* or *custom* encryption with terms such as AES, 3DES, Blowfish etc. What this likely means is that the vendor is using standard cryptographic algorithms mixed up with their own proprietary cryptographic algorithm. Encryption algorithms are just a small part of a cryptographic protocol. You can bet the weak link is the *proprietary* component. **Major Red Flag**.

Question: A vendor claims to use SSH but when I look closely, it does not look like it is being used **END** to **END**.

Answer: Some companies claim to have SSH but when you examine their claim, SSH may only be used within the server but things change to *proprietary* from the server to the devices, which is where the data is most vulnerable. In this case the weakest link is the transmission of data from the server to the device, making the entire solution unsecure. Again, when evaluating security software keep and eye out for the words *end-to-end* and *proprietary* mixed!

Question: A vendor claims to have FIPS 140-2 but they don't have FIPS 140-2 compliant client.

Answer: Again, as unfortunate as it is, some companies claim to have features when they simply just do not. They may be compliant on parts of the server, but if its not FIPS 140-2 compliant on the client then its not compliant **END** to **END**.

Question: Why is Proprietary Encryption a Red Flag?

Answer: Existing Cryptography for our industry is quite good due to dedicated, highly skilled mathematicians and the best cryptographers at security agencies such as the NSA (National Security Agency) and first class universities. Good cryptographic algorithms require complicated mathematics in addition to expensive technologies for development. Algorithm acceptance requires testing and scrutiny of many brilliant people as well as the industry peer review and time in the field.

Commercial software vendors typically venture into the proprietary cryptographic arena to save time or money. A few "sharp" engineers creating a proprietary cryptographic algorithm is not remotely comparable to established cryptographic algorithms standardized by agencies such as the NSA. At best it is arrogant when software vendors believe they can do a better job than the professional cryptographers; at worst customer systems are breached.

Question: Our vendor says that they developed their own cryptographic protocol?

Answer: Run, Run, Run as fast as you can! Encryption protocols are extremely difficult to design and are not for the faint of heart. This is a very dangerous situation because there is a false sense of security. Developers often believe they have correctly implemented a cryptographic protocol or encryption algorithm only to later find out that many significant potential exploits and other security risks exist after many months of deployment. There is no replacement for many years of public scrutiny and testing. Security by obscurity can never work.

Question: Our vendor refuses to give details of their cryptographic protocol design on the grounds that it jeopardizes the security of the solution?

Answer: All standard cryptographic protocols are described in detail on the level of design. Your vendor is trying to achieve security through obscurity. This simply does not work because of all the hardware and software tracing tools available to determined hackers.