SERPENTEQ
Sicherheit ist nur eine Illusion

The SERPENTEQ SAP Security Blog

Heidelberg, May 9, 2019

10KBlaze - Hype & Scaremongering

On April 19, 2019, at the OPCDE Cyber Security conference in Dubai, security researchers Dmitry Chastuhin and Mathieu Geli gave a presentation called “SAP gateway to Heaven[1]”.

They re-visited two configuration issues (related to *SAP Gateway* and *SAP Message Server*) that have been known for many years and for which detailed security guidelines have been available for years. Now the researchers applied some admirably creative thinking to combine them.

They demonstrated that the default configuration of the SAP Gateway’s *secinfo* file can lead to a vulnerability if the SAP Message Server configuration is insecure (which is the case in the default config). Their talk was delivered in great technical detail and they also released python scripts as PoC. So far so good.

Unfortunately, in the wake of this talk, a wave of press releases and a flood of “you are doomed” mails have been unleashed onto companies running SAP by some security companies (which appear to have no part in the research work of Dmitry Chastuhin and Mathieu Geli).

If you read these virtually apocalyptical messages, you may come to think that your company’s SAP servers will be hacked any second via the Internet.

We write this blog post in order to share important facts and reduce the current level of uncertainty. The information in this blog is backed up by experienced experts from two other SAP security companies (EUROSEC and akquinet).

First of all: **if you configure your SAP Gateway securely, even an insecure SAP Message Server configuration will NOT expose your gateway to remote code execution**.

Dmitry Chastuhin and Mathieu Geli correctly state in their talk that the secinfo setting can lead to a vulnerability if the SAP Message Server configuration is insecure. However, the secinfo setting USER-HOST=*LOCAL* is secure, no matter what you do with a SAP Message Server. But for some mysterious reason, there is no trace of this information in any of the press releases and warning mails.

The value *INTERNAL* means that all server instances in the SID cluster (which you can join via SAP Message Server) can connect to the SAP Gateway. This may be a problem. And in most cases customers actually need this setting in productive use.

The value *LOCAL* , however, means that only the network cards of the server instance running the SAP Gateway can connect to its gateway. If you configured your secinfo settings this way, no one can exploit your SAP Gateway over the network. Neither from the outside nor from the inside.

And to make that clear as well, the gateway attack described in the talk requires network access to the SAP Gateway. And unless your admins are reckless daredevils, your gateway is only accessible from within your intranet. Granted, this won’t protect your company from harm, but it keeps the attack surface manageable.

But apart from this, a successful attack requires several other preconditions, as explained in the following.

[1] https://github.com/comaeio/OPCDE/tree/master/2019/Emirates/(SAP)%20Gateway%20to%20Heaven%20-%20Dmitry%20Chastuhin%2C%20Mathieu%20Geli

In the past, all functionality of SAP Message Server was exposed to clients via a single port. This included the capability to register application servers and was a bad idea. SAP changed this behavior years ago. SAP Note 1421005 deals with the secure configuration of SAP Message Server. Since clients need to be able to connect to a SAP Message Server, access to critical functionality (such as server registration) was delegated to a separate port - the *internal port*.
Of course, this port (TCP 39XX) needs to be protected from client access. Otherwise it would make little sense to spilt access.

But even if this port is accessible, the SAP Message Server can still be protected by proper ACL configurations. (See SAP Notes 821875 and 1495075). This ACL configuration is the best line of defense. It should contain a white list of all application server instances that are allowed to access the SAP Message Server. This prevents any access from unwanted systems (with network access).

If you run a SIEM solution, it also makes sense to activate logging on the SAP Message Server, so you can quickly detect attempts to attack you.

Now what exactly is the risk? The following preconditions are required for remote code execution:

1. The SAP Message Server internal port (39XX) is exposed to clients / the intranet.
2. The SAP Message Server ACL is not (securely) configured. Unfortunately this is the SAP default setting.
3. The SAP Gateway secinfo configuration uses USER-HOST= *INTERNAL* or is in itself configured insecurely. In the latter case companies are vulnerable no matter how their SAP Message Server is configured.
4. An attacker needs physical access to the network (unless the gateway is exposed to the Internet).

Only if all four conditions apply, the 10KBlaze attack poses an additional risk to your company.

Another reason to make sure that your SAP Message Server is sufficiently protected is the “bonus attack” discussed in the talk: An untrusted application server - registered via an unprotected message server - would be able to steal login credentials. Unless a company uses SNC (Secure Network Communication).

We worked with some of our customers to cool down the panic-mode caused by the press releases. So far we haven’t seen any additional risks arising from 10KBlaze.

Our advice: if your SAP admins even remotely pay attention to SAP Notes, it is very likely that the exploits shown by the two researchers have no adverse effect. You should check your settings nonetheless. But do it with calm and serenity.

Xu Jia and Andreas Wiegenstein