

strongSwan - Bug #873

RFC 7427 signature authentication and pkcs11 plugin

04.03.2015 11:25 - Luka Logar

Status:	Closed	Start date:	04.03.2015
Priority:	Normal	Due date:	
Assignee:	Tobias Brunner	Estimated time:	0.00 hour
Category:	libstrongswan	Resolution:	Fixed
Target version:	5.3.0		
Affected version:	dr rc master		

Description

Hi, when SIGN_ECDSA_WITH_SHAxxx_DER (and possibly other _SHAxxx_DER) signature scheme is used in combination with pkcs11 plugin, raw output of C_sign() should be asn1 encoded, otherwise signature verification fails.

This ugly patch provides proof of concept code that makes signature authentication work:

```
--- a/src/libstrongswan/plugins/pkcs11/pkcs11_private_key.c
+++ b/src/libstrongswan/plugins/pkcs11/pkcs11_private_key.c
@@ -24,6 +24,8 @@

#include <utils/debug.h>

+#include <asn1/asn1.h>
+
typedef struct private_pkcs11_private_key_t private_pkcs11_private_key_t;

/**
@@ -288,7 +290,11 @@ METHOD(private_key_t, sign, bool,
    free(buf);
    return FALSE;
}
- *signature = chunk_create(buf, len);
+ if ((scheme == SIGN_ECDSA_WITH_SHA1_DER) || (scheme == SIGN_ECDSA_WITH_SHA256_DER) || (scheme
== SIGN_ECDSA_WITH_SHA384_DER) || (scheme == SIGN_ECDSA_WITH_SHA512_DER)) {
+     *signature = asn1_wrap(ASN1_SEQUENCE, "mm", asn1_integer("c", chunk_create(&buf[0], len/2
)), asn1_integer("c", chunk_create(&buf[len/2], len/2)));
+     free(buf);
+ } else
+     *signature = chunk_create(buf, len);
    return TRUE;
}
```

Best regards

Luka

Associated revisions

Revision b258ed01 - 09.03.2015 15:37 - Tobias Brunner

pkcs11: Properly encode RFC 3279 ECDSA signatures

Fixes #873.

Revision e5009fbb - 09.03.2015 15:37 - Tobias Brunner

pkcs11: Convert RFC 3279 ECDSA signatures when verifying

References #873.

History

#1 - 05.03.2015 17:10 - Tobias Brunner

- *Tracker changed from Issue to Bug*
- *Category set to libstrongswan*
- *Status changed from New to Feedback*
- *Assignee set to Tobias Brunner*
- *Target version set to 5.3.0*

Correct, these signature schemes are currently not handled properly by the *pkcs11* plugin. We should also handle this in the *verify()* method of the PKCS#11 public key wrapper, which might get used if the *use_pubkey* option of the *pkcs11* plugin is enabled.

I pushed two fixes to the *pkcs11-ecdsa* branch.

#2 - 06.03.2015 12:37 - Luka Logar

Tobias, thanks. *pkcs11-ecdsa* branch is working just fine.

Regards
Luka

#3 - 09.03.2015 15:39 - Tobias Brunner

- *Status changed from Feedback to Closed*
- *Resolution set to Fixed*

Thanks for testing. Merged to master.