p1b19i1187s0-.pdf

Unknown

From: Alan Guggenheim [aguggenheim@cis-tech.com]
Sent: Thursday, October 14, 1999 5:24 PM
To: Christophe Gaillard (E-mail); Christophe Nicolas (E-mail); Henri Kudelski (E-mail); Jean-Daniel Meynet (E-mail); Marco Sasselli (E-mail); Olivier Brique (E-mail)
Subject: Forum 566

Importance: High

*** PGP Signature Status: good
*** Signer: Alan A. Guggenheim <guggenheim@nagra.com>
*** Signed: 10/14/99 6:23:10 PM
*** Verified: 10/14/99 9:49:52 AM
*** BEGIN PGP DECRYPTED/VERIFIED MESSAGE ***

Author Topic: Tech info on dump fragments
Code

Member
Total posts:
23
Date Joined:
10-01-1999 posted October 13, 1999 11:36 AM

I looked over the pieces on the net. There is more than 7f00 that is mixed from rom 2 and rom 3. It was probably done by accident and this is just for your information.
PS- there is no A0 C0 command but if you can sign any packet out there, sign one that performs an eeprom write via the f0/f3 command since NAgra doesnt require any kind of security pre-checks other than a legit signature. Common sig is valid people!

1 emm with F0/F3 inside can write you a door into the card.

IP: Logged

DataShark

Member
Total posts:
22
Date Joined:
08-31-1999 posted October 13, 1999 11:50 AM

Whew! I was wondering if I had got the disassembly wrong. This just confirms what I've been thinking. I guess I'll see if any more commands I can turn inside out.
DS

IP: Logged

ukapache

Member
Total posts:
14
Date Joined:

CASE NO.
SA CV 03-950 DOC (JTLx)
ECHOSTAR SATELLITE CORP., et al.,

vs.

NDS GROUP PLC, et al.

DEFENDANT'S EXHIBIT 1202

DATE _____ IDEN.

DATE _____ EVID.

BY _____
Deputy Clerk

p1b19i1187s0-.pdf

Unknown

From: Alan Guggenheim [aguggenheim@cis-tech.com]
Sent: Thursday, October 14, 1999 5:24 PM
To: Christophe Gaillard (E-mail); Christophe Nicolas (E-mail); Henri Kudelski (E-mail); Jean-Daniel Meynet (E-mail); Marco Sasselli (E-mail); Olivier Brique (E-mail)
Subject: Forum 566

Importance: High

*** PGP Signature Status: good
*** Signer: Alan A. Guggenheim <guggenheim@nagra.com>
*** Signed: 10/14/99 6:23:10 PM
*** Verified: 10/14/99 9:49:52 AM
*** BEGIN PGP DECRYPTED/VERIFIED MESSAGE ***

Author Topic: Tech info on dump fragments
Code

Member
Total posts:
23
Date Joined:
10-01-1999 posted October 13, 1999 11:36 AM

I looked over the pieces on the net. There is more than 7f00 that is mixed from rom 2 and rom 3. It was probably done by accident and this is just for your information.
PS- there is no A0 C0 command but if you can sign any packet out there, sign one that performs an eeprom write via the f0/f3 command since NAgra doesnt require any kind of security pre-checks other than a legit signature. Common sig is valid people!

1 emm with F0/F3 inside can write you a door into the card.

IP: Logged

DataShark

Member
Total posts:
22
Date Joined:
08-31-1999 posted October 13, 1999 11:50 AM

Whew! I was wondering if I had got the disassembly wrong. This just confirms what I've been thinking. I guess I'll see if any more commands I can turn inside out.
DS

IP: Logged

ukapache

Member
Total posts:
14
Date Joined:

09-13-1999 posted October 13, 1999 01:53 PM

To add to Code's post, here's the hash routine again...
Given 5 8-byte blocks b1 to b5
r1 = enc(b1,verify_key)
r2 = enc(b2, r1 xor b1)
r3 = enc(b3, r2 xor b2)
r4 = enc(b4, r3 xor b3)
valid_hash=enc(b5, r4 xor b4)

enc is the same as encrypt for control words with boxkey in xfile

i hope this info is correct since I dont have time to check right now... i'll edit if faulty but it should be right.

IP: Logged

DataPimp

Member
Total posts:
11
Date Joined:
09-12-1999 posted October 13, 1999 04:19 PM

Can you tell us what you mean by common sig? and secondly, whats the format of the f0 command?
IP: Logged

StuntGuy

Member
Total posts:
27
Date Joined:
09-02-1999 posted October 13, 1999 05:23 PM

DP: The format of the F0/F3 command is:
F0/F3 <6805 object code here> 20 ff

(the 20 ff at the end is a BRA to itself, which will cause the card to hang after your code has finished, preventing it from executing code off into the weeds someplace and potentially damaging it.)

When the card is processing the 00 command and it determines that the next subcommand to process (which is always at location $80) is F0 (for 288-01 cards) or F3 (for 288-02 cards), it handles it by doing a JMP $81, which causes the code at <6805 object code here> to be executed.

Something else you could do if you can sign (and encrypt) a 00 packet is send an F0/F3 command that would dump the card (except for ROM from $2000..$3FFF- The guys at ST micro understand the use of the MAC matrix...the guys at Nagra apparantly either don't, or don't care.)

-s

[This message has been edited by StuntGuy (edited October 13, 1999).]

IP: Logged

GreyCat

2

Member
Total posts:
11
Date Joined:
09-17-1999 posted October 13, 1999 08:55 PM

StuntGuy,
Why would you need to sign and encrypt the to dump the ROM?

Can you elaborate a little?

IP: Logged

bplus

Member
Total posts:
3
Date Joined:
09-22-1999 posted October 13, 1999 09:16 PM

Just had a thought ...if we had knowledge of the verify keys in the card..could we build a packet to get inside the card and dump the eeprom contents?
If this is the case..seems to me we already know what the verify key is in a SV card and this could possibly be applied to a SV subbed cam .
IP: Logged

M_DeRuKe

Elite Member
Total posts:
206
Date Joined:
09-05-1999 posted October 14, 1999 12:47 AM

Code, are there different versions of what should be the same ROM in the cards? (are there cards that have different roms?) A few months back, there were a lot of "this receiver is not valid with this receiver" or to that effect. Is this the resultant of the kill command of JMP E186 that SG was talking about? If not, would the receiver give you an error message as to an invalid card? Or were all those errors physical errors in card that isn't code related?
Thanks

IP: Logged

StuntGuy

Member
Total posts:
27
Date Joined:
09-02-1999 posted October 14, 1999 07:54 AM

GC: Because the data that is interpreted as subcommands for the 00 packet is sent in a 64-byte encrypted data block, and is preceded by an 8-byte signature which is based on the decrypted data. (Note: the remainder of this paragraph is speculative, because so far, I don't think I have the ROM code that deals with this, so I can't say for SURE what happens, but assuming that the Nagra guys were even a tiny bit competent, this is what _should_ happen.) When the card receives a 00 packet, is first decrypts the 64 bytes of encrypted data, then is computes what it believes should be the correct signature based on the decrypted result. If the signature matches, the card then executes the subcommands stored within the now-decrypted data.
BP: It's not just a matter of having the verify key. That would help compute the valid

signature, but we also need to know the keys that are used to encrypt the 64 bytes of EMM data itself as well as the algorithm that was used for the encryption.

M_D: Yes, there are cards that have different ROMs. If you look at the back of your E* cards, holding them so that the contacts are at the top, you'll see a little number printed in the bottom-right area of the card. For early E* cards, this number is 288-01. The ROMs in this card were quite buggy, and as a consequence, there are a lot of fixes to the ROM code stored in the EEPROM (about 900 hex bytes, in fact). The later versions will say "288-02". These cards have basically the same code, but with the fixes that were implemented in the 288-01's EEPROM integrated back into the ROM. If you download the E* card dumps from here, the REV052 dump is the EEPROM from a 288-01 card, while the REV313 and REV367 dumps are from a 288-02 card. As a result of the two ROMs being different (the code segments that were fixed ended up being different lengths than the original code), and because E* wanted to have both cards be usable, the 288-02 cards use a different "execute code" EMM subcommand than the 288-01 cards: For the 288-01 cards, the "execute code" subcommand is F0, while on the 288-02 cards, it's F3. This allows E*/Nagra to send "execute code" subcommands that will be executed on the 288-01s (and be able to call their ROM routines) without the danger of a 288-02 card attempting to execute that code, and vice-versa. My understanding is that SkyVista and ExVu and all of the other cards that deal with all of the other services that E* provides are based on the same ROM as the 288-02 E* cards, but they have a different 288-xx number than the E* cards (288-03, 288-04, etc.)

All: Just to clear up any possible confusion, the ROM images that have been released on the net are _mostly_ from a 288-01 card's ROM image (we refer to this image as ROM2). The notable exception, as Code mentioned, is a section that ranges from 7F00..7FEF, which was released by Macro. This section came from a 288-02 ROM image (which we refer to as ROM3). There was a version of 7F00..7FFF released by xchi that came from a ROM2 image, though...the one you want is the one that has the values 20 0B AE at 7F00. What does this mean to you? It means that a whole bunch of the ROM is available on the 'net, because keep in mind that for the most part, the code in the EEPROM is basically replacement code for some of what's in the ROM. Therefore, between the various snippets of ROM that have been released, as well as the REV052 EEPROM dump, there's quite a bit of data available. And yes, this also means that I'm now pretty comfortable saying that the ROM images that have been released are either legitimate or a very clever (and painstakingly crafted) fake.

-s

[This message has been edited by StuntGuy (edited October 14, 1999).]

Alan A. Guggenheim
Senior Vice President
Strategic Business Development
guggenheim@nagra.com <mailto:guggenheim@nagra.com>
Tel: +1 (310) 967-7770 xt 501

*** END PGP DECRYPTED/VERIFIED MESSAGE ***