Säkerhet i industriella informations- och styrsystem

Nationellt program för ökad säkerhet i industriella informations- och styrsystem syftar till att öka den nationella förmågan att hantera IT-relaterade hot mot s.k. SCADA-system - industriella informations- och styrsystem i samhällsviktiga verksamheter och kritisk infrastruktur. Programmets mål är att höja den tekniska kompetensen, sprida information och praktiskt stödja användare av SCADA-system för att öka samhällets säkerhet. Programmet drivs av MSB i samarbete med ett stort antal offentliga och privata aktörer.

FOI leder på MSB:s uppdrag programområde 1, Teknisk samverkansplattform. Det omfattar ett avancerat SCADA-laboratorium med demonstratorer, kompetensuppbyggnad, kurser i IT-säkerhet, nationella och internationella övningar samt forskningssamarbete.

FOI
Totalförsvarets forskningsinstitut
164 90 STOCKHOLM

Telefonväxel 08 555 030 00
Fax 8 555 031 00

www.foi.se

Myndigheten för samhällsskydd och beredskap
651 81 KARLSTAD

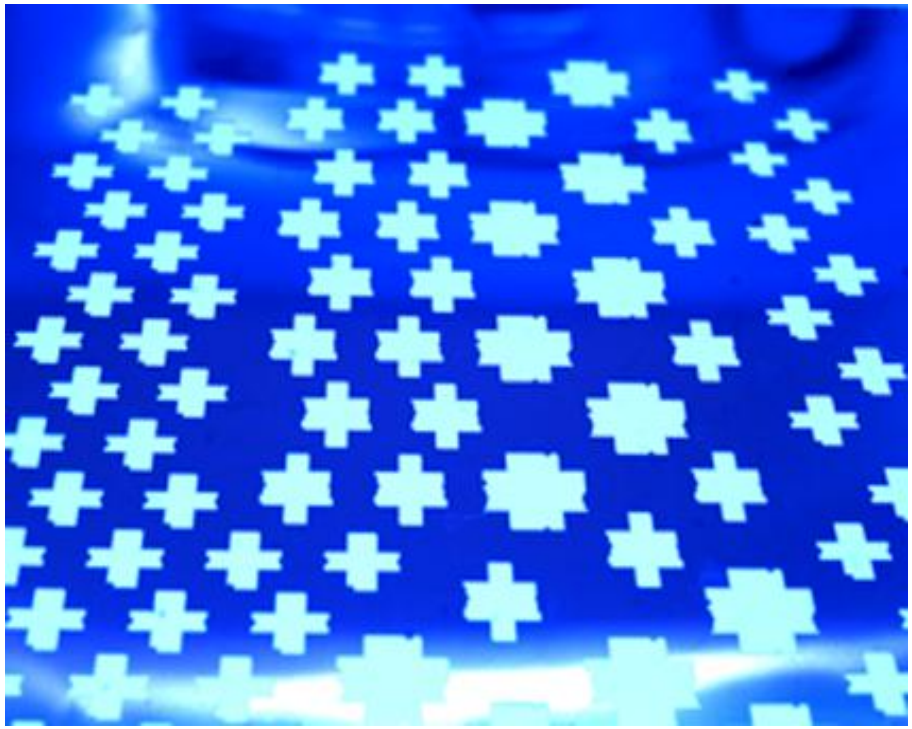
Telefonväxel: 0771-240 240
Fax: 010-240 56 00

www.msb.se

Möjligheter och problem vid analys av fientlig kod riktad mot Siemens S7-serie

ARNE VIDSTRÖM

FOI
MSB



FOI-R--3567--SE
ISSN 1650-1942

December 2012

Arne Vidström

Möjligheter och problem vid analys av fientlig kod riktad mot Siemens S7-serie

Titel	Möjligheter och problem vid analys av fientlig kod riktad mot Siemens S7-serie
Title	Opportunities and problems in the analysis of malware directed against Siemens S7
Rapportnr/Report no	FOI-R--3567--SE
Månad/Month	November
Utgivningsår/Year	2012
Antal sidor/Pages	35 p
ISSN	1650-1942
Kund/Customer	MSB
FoT område	
Projektnr/Project no	E323125
Godkänd av/Approved by	Christian Jönsson
Ansvarig avdelning	Informations- och areosystem

Sammanfattning

Den här rapporten beskriver möjligheter och problem vid analys av fientlig kod riktad mot Siemens S7-serie av PLC:er (Programmable Logic Controller). Fokus ligger på odokumenterad funktionalitet eftersom sådan utgör ett av de främsta problemen vid analysen innan man har skaffat sig den kunskap som krävs. Det har visat sig vara möjligt att få fram en del allmän information från block av fientlig kod. Det är också fullt möjligt att identifiera vilken/vilka plattformar sådana block riktar sig mot baserat på utseendet hos maskinkoden, även när det gäller fullständigt odokumenterade plattformar. Det öppnar upp en möjlighet att till exempel konstruera automatiserade verktyg för plattforms-identifikation. Däremot är det mer komplicerat att lista ut exakt vad fientlig kod gör, även om det tycks finnas framkomliga vägar även på det området. Sammanfattningsvis kan sägas att vi i dagsläget har nått en jämförelsevis hög kunskapsnivå när det gäller hur Siemens S7-serie fungerar i teknisk detalj. Den här typen av kunskap är generellt sett begränsad till vissa utvecklare av PLC:er inom Siemens, samt till enstaka experter utanför Siemens.

Nyckelord: Siemens, S7-300, S7-400, S7-1200, MC7, STL, S7-protokollet, Stuxnet, fientlig kod

Summary

This report describes possibilities and obstacles related to the analysis of malware that targets the Siemens S7 series of PLCs (Programmable Logic Controller). The focus is on undocumented functionality since it is one of the largest obstacles in such analysis before one has obtained the relevant knowledge. Retrieving general information from blocks of malware turned out to be possible. It is also possible to identify which platform(s) such blocks are targeting based on general features of the machine code, even on completely undocumented platforms. This also makes it possible to construct automated tools for platform identification. On the other hand, it is more complicated to figure out exactly what the malware does, even though there seems to be ways forward in that area too. Overall we have reached a comparatively high level of knowledge when it comes to the low level functionality in Siemens S7. This kind of knowledge is generally limited to some developers of PLCs within Siemens, and to a few experts outside of Siemens.

Keywords: Siemens, S7-300, S7-400, S7-1200, MC7, STL, S7 protocol, Stuxnet, malware

Innehållsförteckning

1 Inledning 7

2 Teknisk översikt över Siemens S7-serie 9

2.1 Modellöversikt 9

2.2 Kommunikationsprotokoll över Ethernet 9

2.3 Maskinkoden i S7-400 och S7-1200 11

3 MC7, STL och relationen mellan dem 12

4 S7-protokollet (S7 400) 14

5 Lärdomar från studierna av Siemens S7-serie 16

6 Appendix A – Sekvens 0 från Stuxnet 17

7 Appendix B – Sekvens 1 från Stuxnet 22

8 Appendix C – Sekvens C från Stuxnet 27

1 Inledning

Sedan 2007 har FOI arbetat på uppdrag av MSB med att genomföra verksamhet relaterad till säkerhet i industriella informations- och styrsystem. Uppdragen har bestått av såväl direkt stöd till MSB:s program (från 2009 och framåt) som tekniskt inriktad verksamhet avseende IT-säkerhet. Under 2010 etablerades namnet för ett teknisk centrum; Nationellt centrum för säkerhet i styrsystem för samhällsviktig verksamhet (NSC3).

Den här rapporten beskriver möjligheter och problem vid analys av fientlig kod riktad mot Siemens S7-serie av PLC:er (Programmable Logic Controller). Fokus ligger på odokumenterad funktionalitet eftersom sådan utgör ett av de största problemen vid analysen innan man har skaffat sig den kunskap som krävs.

Den främsta målgruppen för rapporten är tekniker som har generella kunskaper om analys av fientlig kod och tillgång till den officiella dokumentationen från Siemens. En annan målgrupp är personer som har ett mer generellt intresse för området. I första hand rekommenderar jag att de sistnämnda läser kapitel fem, som handlar om lärdomar från studierna av S7.

Informationen i rapporten bygger på den förståelse som jag har byggt upp genom egna praktiska experiment med S7. Som stöd för mitt eget experimenterande har jag självklart också införskaffat information från andra källor, men ofta har den varit fragmentarisk och inte helt tillförlitlig. Eftersom innehållet i rapporten rör odokumenterad funktionalitet så finns en uppenbar risk att felaktigheter har smugit sig in även här. I första hand har jag försökt förhindra det genom att så långt som möjligt undersöka allt praktiskt på egen hand istället för att förlita mig på andra källor.

Slutligen bör nämnas att den tekniska detaljnivån har valts delvis med tanke på att rapporten behandlar odokumenterad funktionalitet. Jag har försökt få med tekniskt användbar information utan att för den skull lämna ut information som Siemens kan uppleva som känslig. Projektet har alltså lett fram till mer detaljerade kunskaper än de som framkommer i den här rapporten. Dessa kan vi självklart också

använda oss av vid eventuella framtida incidenter som involverar fientlig kod riktad mot Siemens S7.

2 Teknisk översikt över Siemens S7-serie

2.1 Modellöversikt

Det finns fyra övergripande modeller av PLC:er i Siemens S7-serie:

- ✓ S7-200 är en äldre och enklare modell
- ✓ S7-300 är mellanmodellen
- ✓ S7-400 är den kraftfullaste modellen
- ✓ S7-1200 är en nyare och enklare modell som ska ersätta S7-200

Eftersom jag bara har haft tillgång till S7-400 och S7-1200 så är det de modellerna som behandlas närmare i den här rapporten. S7-300 liknar antagligen S7-400 arkitekturmässigt i hög grad, men utan tillgång till en S7-300 är det svårt att uttala sig om hur lika eller olika de egentligen är. En skillnad är till exempel att S7-400 har två extra ackumulatorer (ACCU 3 och ACCU 4) jämfört med S7-300, men i övrigt ser de ut att ha samma registeruppsättning.

2.2 Kommunikationsprotokoll över Ethernet

Både S7-400 och S7-1200 använder protokoll i flera lager över Ethernet. Direkt över Ethernet ligger de välkända protokollen IP och TCP. Ovanpå TCP finns ett tunt lager som kallas ISO-TSAP (ISO Transport Services Access Protocol). ISO-TSAP gör egentligen inget mer än att klumpa ihop en kontinuerlig ström av bytes (TCP-ström) till diskreta grupper av bytes (TPDU = Transport Protocol Data Unit). Den port som reserverats för protokollet är TCP 102.

Ovanpå ISO-TSAP ligger protokollet ISO 8073, som också kallas COTP (Connection Oriented Transport Protocol). ISO 8073 kan betraktas som motsvarigheten till TCP i OSI-protokollen.[1] Eftersom

[1] OSI-protokollen är en grupp av kommunikationsprotokoll som utvecklades på 1970-talet av ISO (International Organization for Standardization) och ITU-T (International Telecommunication

ISO 8073 förutsätter att underliggande protokoll delar upp strömmen av data i diskreta grupper av bytes så krävs ISO-TSAP som ett mellanlager mellan TCP och ISO 8073. Ovanpå ISO 8073 ligger Siemens proprietära S7-protokoll, som är designat för att ligga just ovanpå ISO 8073. ISO-TSAP och ISO 8073 fyller alltså egentligen ingen annan funktion än att möjliggöra användning av S7 över TCP.

Flera källor[2] [3], inklusive ICS-CERT, beskriver ISO-TSAP som ett gammalt klartextprotokoll som har orsakat flera av de uppmärksammade säkerhetsproblemen hos Siemens PLC:er. Det är egentligen felaktigt, och kan möjligen bero det på att de inte ens känner till att det ligger ytterligare protokoll (S7) ovanpå ISO-TSAP. Flera källor uttrycker sig nämligen som om ISO-TSAP spelade samma roll som ISO-TSAP, ISO 8073 och S7 gör tillsammans. Förklaringen är förmodligen att S7-protokollet är odokumenterat, vilket gör att ”det sista man ser” när man tittar på trafiken är de båda ISO-protokollen och då främst ISO-TSAP. I själva verket är S7 ett relativt omfattande protokoll. Det innehåller bland annat stöd för autentisering, men med mycket dålig utformning rent säkerhetsmässigt (vilket behandlas närmare senare).

Ovanför ISO 8073 skiljer sig protokollen åt för S7-400 och S7-1200. Det finns också skillnader mellan olika versioner av de båda protokollen. Alla dessa varianter är proprietära och odokumenterade – Siemens har alltså inte släppt någon av specifikationerna offentligt. Protokollet för S7-400 har däremot analyserats av utomstående så långt att stora delar är kända, medan protokollet för S7-1200 är helt okänt utöver sitt namn.

Union - Telecommunication Standardization Sector). OSI är förkortning för Open Systems Interconnection.

[2] ICS-CERT, *ICSA-11-223-01 - A summary of reported issues affecting Siemens Simatic PLCs*, 2011-08-11, hämtad 2012-09-20, <http://www.us-cert.gov/control_systems/pdf/ICSA-11-223-01.pdf>

[3] Beresford, D. *Exploiting Siemens Simatic S7 PLCs*, 2011

2.3 Maskinkoden i S7-400 och S7-1200

S7-400 (och även S7-300) använder en maskinkod som kallas MC7. Den är proprietär och odokumenterad, precis som S7-protokollet. Utomstående har lyckats analysera och dokumentera delar av MC7. Den något skissartade dokumentationen är tyvärr långt ifrån komplett. Dessutom innehåller den flera felaktigheter, vilket framkom när jag själv undersökte förhållandet mellan MC7 och STL (STatement List) närmare.

S7-1200 använder en annan typ av maskinkod än MC7, men därutöver är inget annat känt om den vare sig publikt eller av mig. Eftersom det inte finns något stöd för motsvarigheten till assembler för S7-1200 så är det mycket svårare att lista ut hur maskinkoden fungerar där.

3 MC7, STL och relationen mellan dem

Motsvarigheten till assembler i S7 kallas STL. Som exempel på MC7 och STL följer ett slumpmässigt valt utdrag av kod ur Stuxnet.

7E 62 00 02	L LW 2	Lagra i lokalt word 2
68 1E	ITD	Omvandla 16-bitars heltal till 32-bitars heltal
7E 67 00 10	T LD 16	Överför till lokalt dword 16
70 02	TAK	Växla innehållet i ACCU 1 och ACCU 2
38 03 00 00 10 FF	L L#4351	Ladda 4351 som 32-bitars tal in i ACCU 1
60 0D	+D	Addera ACCU 1 och ACCU 2 och lagra I ACCU 1
7E 63 00 10	L LD 16	Överför till lokalt dword 16
60 0D	+D	Addera ACCU 1 och ACCU 2 och lagra i ACCU 1
68 1D	SET	Sätt RLO till 1
FE 04	LAR1	Ladda ACCU 1 i AR1
79 D2 00 00	= DIX [AR1, P#0.0]	Tilldela till bit [AR1+0] i öppet datainstansblock

Till vänster finns MC7-koden i hexadecimalt format. I mitten finns manuellt översatt STL-kod och till höger en kortfattad beskrivning av vad varje instruktion gör. När man får tag på ett exemplar av fientlig kod är det den vänstra kolumnen man har tillgång till (antingen direkt eller efter exempelvis dekryptering). Därifrån måste man göra översättningen till den mellersta kolumnen och utifrån det lista ut vad koden innebär (den högra kolumnen). Slutligen behöver man förstå vad den högra kolumnen innebär i ett större sammanhang – alltså vad koden gör med styrsystemet som helhet.

Hur man programmerar i STL är officiellt dokumenterat. Det som inte är dokumenterat är maskinkoden (MC7) till vänster, samt hur man gör översättningar fram och tillbaka mellan den och STL. Med andra ord krävs en ganska specialiserad kompetens för att kunna överlämna begriplig kod till en styrsystemsexpert som sedan kan lista ut vad den gör med styrsystemet som helhet. En alternativ väg är att bygga speciella verktyg som kan placera in fientlig kod i rätt position i Siemens STEP7-mjukvara och låta den utföra översättningen från

MC7 till STL.[4] Det finns däremot inget inbyggt stöd för att göra det på ett enkelt sätt.

[4] Jag har ännu inte undersökt möjligheten närmare själv, men det lär vara möjligt bland annat enligt Felix Lindner (personlig kommunikation).

4 S7-protokollet (S7 400)

För att förstå principen bakom S7-protokollet kan ett par illustrativa exempel vara användbara.

Mjukvaran STEP7 i den dator som kopplas till PLC:n kan till exempel be om att få läsa något av innehållet i systemstatuslistan (SSL). Då skickar den ett paket till PLC:n där den anger att det rör sig om en förfrågan från funktionsgruppen SSL-funktioner och mer specifikt underfunktionen att läsa SSL. Dessutom anger den vilket SSL-ID och index den vill läsa. PLC:n skickar i sin tur tillbaka ett paket där den anger att det rör sig om ett svar från funktionsgruppen SSL-funktioner, och mer specifikt underfunktionen att läsa SSL. Den anger också vilket SSL-ID och index det rör sig om. Sedan följer just detta innehåll. Ett exempel på information som kan efterfrågas är vilken skyddsnivå (0-3) som PLC-CPU: n befinner sig i.

Om PLC:n till exempel befinner sig i skyddsnivå 3 så krävs ett lösenord för att bland annat ladda upp ett kodblock till den. STEP7 skickar då ett paket där den anger att det rör sig om en förfrågan från funktionsgruppen säkerhet och mer specifikt underfunktionen för att ange lösenord. Sedan följer lösenordet i förvrängd form. Precis som i exemplet med systemstatuslistan ger PLC:n ett svar - i det här fallet får STEP7 veta om lösenordet var det rätta eller inte.

Den förvrängda formen av lösenordet visade sig vid närmare undersökning bygga på en algoritm som går att köra baklänges (åtminstone för S7-400). Det innebär alltså att den som kan avlyssna nätverkstrafiken när lösenordet anges också kan räkna fram lösenordet i klartext på ett ögonblick. Säkerhetsbristens existens visade sig redan ha dokumenterats av Siemens, men utan närmare detaljer.[5] Eftersom jag har listat ut algoritmen i detalj så har jag kunnat konstruera ett Perlscript för snabb avkodning.

[5] Siemens, *Potential Password Security Weakness in SIMATIC Controllers*, 2011-07-05, hämtad 2012-09-26, <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=51401544>

Liknande förfrågningar och svar skickas när man laddar ner ett kodblock till PLC:n. Ett sådant kodblock innehåller dels olika sorters blockinformation och dels maskinkod (MC7). Blockinformationen anger vilket språk koden har kompilerats från, tidsstämplar från kompileringen, samt flera andra uppgifter. Den exakta utformningen hos dessa data ser olika ut beroende på var man tittar på dem i systemet. Till exempel har kodblocken från Stuxnet information i ett annat format än det som används när blocken skickas via S7-protokollet. I appendix A-B finns avkodad information från Stuxnet som exempel på vilken information som går att få fram om man känner till exakt hur den odokumenterade kodningen går till.

När en användare gör något i STEP7-mjukvaran så kan det motsvara en hel serie av olika sorters förfrågningar och svar i S7-protokollet. Till exempel involverar uppladdning av ett kodblock till en lösenordsskyddad PLC alla de tre typerna av kommunikation som beskrivits ovan. För användaren motsvarar det däremot bara ett enda klick på ”Download” i en meny (om lösenordet redan har angivits tidigare vill säga).

5 Lärdomar från studierna av Siemens S7-serie

Studierna av Siemens S7-serie har tydliggjort både möjligheter och problem vid analys av fientlig kod riktad mot industriella styrsystem.

Det är fullt möjligt att få fram en del allmän information från block av fientlig kod (se appendix A-C för exempel på vad man kan få fram). Det är också fullt möjligt att identifiera vilken/vilka plattformar sådana block riktar sig mot baserat på utseendet hos maskinkoden, även när det gäller fullständigt odokumenterade plattformar som S7-1200. Det öppnar upp en möjlighet att bygga automatiserade verktyg för plattformsidentifikation, alternativt att bygga upp kompetens för manuell identifikation. Det sistnämnda är ändå ett grundläggande steg för att kunna bygga automatiserade verktyg. För att nå dit krävs i sin tur motsvarande studier av fler tillverkares system än Siemens. Med hjälp av ett sådant verktyg skulle man vid upptäckt av ny fientlig kod kunna peka ut plattformar som löper risk att drabbas.

Däremot är det mycket svårare att bygga upp förmåga att *tolka* maskinkod till S7-1200 på grund av att den plattformen saknar stöd för STL. Den lärdomen är förmodligen överförbar till system från andra tillverkare, åtminstone i den mån det finns sådana system med motsvarande förutsättningar. När det gäller S7-400 är det möjligt att tolka en avsevärd del av maskinkoden, men i dagsläget går det inte att tolka all maskinkod. Eftersom behovet av verktyg för detta är starkt begränsat finns det inga publikt tillgängliga verktyg från Siemens. Det är också oklart i vilken utsträckning Siemens själva har tillgång till den sortens verktyg. En möjlighet kan vara att bygga speciella verktyg för att kunna ladda in godtycklig fientlig kod i STEP7 och låta översättningen till STL ske där.

Sammanfattningsvis kan sägas att vi i dagsläget har nått en jämförelsevis hög kunskapsnivå när det gäller hur Siemens S7-serie fungerar i teknisk detalj. Den här typen av kunskap är generellt sett begränsad till vissa utvecklare av PLC:er inom Siemens, samt till enstaka experter utanför Siemens.

6 Appendix A – Sekvens 0 från Stuxnet

block_0_type_c.bin

FC 2

Function family: CP_300

Function name: DP_RECV

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 15:00:27 (local time)

Time stamp #2: måndag 2007-09-24 15:00:27 (local time)

block_1865_type_c.bin

FC 1865

Function family: IEC

Function name: S7_LV

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 15:00:34 (local time)

Time stamp #2: måndag 2007-09-24 15:00:34 (local time)

block_1866_type_c.bin

FC 1866

Function family: IEC

Function name: WE_TE

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 15:00:32 (local time)

Time stamp #2: måndag 2007-09-24 15:00:32 (local time)

block_1867_type_c.bin

FC 1867

Function family: IEC

Function name: RF_GH
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 15:00:21 (local time)
Time stamp #2: måndag 2007-09-24 15:00:21 (local time)

block_1868_type_c.bin
FC 1868
Function family: IEC
Function name: AD_TT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 15:00:29 (local time)
Time stamp #2: måndag 2007-09-24 15:00:29 (local time)

block_1870_type_c.bin
FC 1870
Function family: IEC
Function name: HA_FO
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 14:59:38 (local time)
Time stamp #2: måndag 2007-09-24 14:59:38 (local time)

block_1871_type_c.bin
FC 1871
Function family: IEC
Function name: DR_RN
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 15:00:30 (local time)
Time stamp #2: måndag 2007-09-24 15:00:30 (local time)

block_1873_type_c.bin

FC 1873
Function family: IEC
Function name: S7_WO
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 15:00:23 (local time)
Time stamp #2: måndag 2007-09-24 15:00:23 (local time)

block_1874_type_c.bin
FC 1874
Function family: IEC
Function name: ADD_AC
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 14:59:59 (local time)
Time stamp #2: måndag 2007-09-24 14:59:59 (local time)

block_1876_type_c.bin
FC 1876
Function family: CP_300
Function name: DP_SEND
Created in: STL (STatement List)
Time stamp #1: fredag 2006-05-05 14:52:53 (local time)
Time stamp #2: tisdag 1996-01-02 00:05:56 (local time)

block_1877_type_c.bin
FC 1877
Function family: IEC
Function name: RT_OS
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 15:00:25 (local time)
Time stamp #2: måndag 2007-09-24 15:00:25 (local time)

block_1878_type_c.bin
FC 1878
Function family: IEC
Function name: SB_DT_TM
Created in: STL (STatement List)
Time stamp #1: fredag 2002-02-15 16:49:52 (local time)
Time stamp #2: fredag 2002-02-15 16:49:52 (local time)

block_1879_type_c.bin
FC 1879
Function family: IEC
Function name: EQ_DT
Created in: STL (STatement List)
Time stamp #1: fredag 2002-02-15 16:49:58 (local time)
Time stamp #2: fredag 2002-02-15 16:49:58 (local time)

block_1880_type_c.bin
FC 1880
Function family: IEC
Function name: SB_DT_DT
Created in: STL (STatement List)
Time stamp #1: fredag 2002-02-15 16:49:37 (local time)
Time stamp #2: fredag 2002-02-15 16:49:37 (local time)

block_888_type_a.bin
DB 888
Function family:
Function name:
Created in: DB (Data Block)

Time stamp #1: måndag 2007-09-24 14:57:49 (local time)
Time stamp #2: måndag 2007-09-24 14:57:49 (local time)

block_889_type_a.bin
DB 889
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 14:57:52 (local time)
Time stamp #2: måndag 2007-09-24 14:57:52 (local time)

block_890_type_a.bin
DB 890
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 14:57:54 (local time)
Time stamp #2: måndag 2007-09-24 14:57:54 (local time)

block_891_type_a.bin
DB 891
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 14:57:50 (local time)
Time stamp #2: måndag 2007-09-24 14:57:50 (local time)

7 Appendix B – Sekvens 1 från Stuxnet

block_0_type_c.bin

FC 2

Function family: CP_300

Function name: DP_RECV

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 16:01:16 (local time)

Time stamp #2: måndag 2007-09-24 16:01:16 (local time)

block_1865_type_c.bin

FC 1865

Function family: IEC

Function name: S7_LV

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 16:01:23 (local time)

Time stamp #2: måndag 2007-09-24 16:01:23 (local time)

block_1866_type_c.bin

FC 1866

Function family: IEC

Function name: WE_TE

Created in: SCL (Structured Control Language)

Time stamp #1: måndag 2007-09-24 16:01:21 (local time)

Time stamp #2: måndag 2007-09-24 16:01:21 (local time)

block_1867_type_c.bin

FC 1867

Function family: IEC

Function name: RF_GH
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 16:01:10 (local time)
Time stamp #2: måndag 2007-09-24 16:01:10 (local time)

block_1868_type_c.bin
FC 1868
Function family: IEC
Function name: AD_TT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 16:01:18 (local time)
Time stamp #2: måndag 2007-09-24 16:01:18 (local time)

block_1870_type_c.bin
FC 1870
Function family: IEC
Function name: HA_FO
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 16:00:27 (local time)
Time stamp #2: måndag 2007-09-24 16:00:27 (local time)

block_1871_type_c.bin
FC 1871
Function family: IEC
Function name: DR_RN
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 16:01:19 (local time)
Time stamp #2: måndag 2007-09-24 16:01:19 (local time)

block_1873_type_c.bin

FC 1873
Function family: IEC
Function name: S7_WO
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 16:01:12 (local time)
Time stamp #2: måndag 2007-09-24 16:01:12 (local time)

block_1874_type_c.bin
FC 1874
Function family: IEC
Function name: ADD_AC
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 16:00:47 (local time)
Time stamp #2: måndag 2007-09-24 16:00:47 (local time)

block_1876_type_c.bin
FC 1876
Function family: CP_300
Function name: DP_SEND
Created in: STL (STatement List)
Time stamp #1: fredag 2006-05-05 14:52:53 (local time)
Time stamp #2: tisdag 1996-01-02 00:05:56 (local time)

block_1877_type_c.bin
FC 1877
Function family: IEC
Function name: RT_OS
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 16:01:14 (local time)
Time stamp #2: måndag 2007-09-24 16:01:14 (local time)

block_1878_type_c.bin

FC 1878

Function family: IEC

Function name: SB_DT_TM

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:52 (local time)

Time stamp #2: fredag 2002-02-15 16:49:52 (local time)

block_1879_type_c.bin

FC 1879

Function family: IEC

Function name: EQ_DT

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:58 (local time)

Time stamp #2: fredag 2002-02-15 16:49:58 (local time)

block_1880_type_c.bin

FC 1880

Function family: IEC

Function name: SB_DT_DT

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:37 (local time)

Time stamp #2: fredag 2002-02-15 16:49:37 (local time)

block_888_type_a.bin

DB 888

Function family:

Function name:

Created in: DB (Data Block)

Time stamp #1: måndag 2007-09-24 15:58:38 (local time)
Time stamp #2: måndag 2007-09-24 15:58:38 (local time)

block_889_type_a.bin
DB 889
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 15:58:42 (local time)
Time stamp #2: måndag 2007-09-24 15:58:42 (local time)

block_890_type_a.bin
DB 890
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 15:58:43 (local time)
Time stamp #2: måndag 2007-09-24 15:58:43 (local time)

block_891_type_a.bin
DB 891
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 15:58:39 (local time)
Time stamp #2: måndag 2007-09-24 15:58:39 (local time)

8 Appendix C – Sekvens C från Stuxnet

block_6055_type_c.bin

FC 6055

Function family: IEC

Function name: SB_DT_TM

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:52 (local time)

Time stamp #2: fredag 2002-02-15 16:49:52 (local time)

block_6056_type_c.bin

FC 6056

Function family: IEC

Function name: SB_DT_DT

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:37 (local time)

Time stamp #2: fredag 2002-02-15 16:49:37 (local time)

block_6057_type_c.bin

FC 6057

Function family: IEC

Function name: EQ_DT

Created in: STL (STatement List)

Time stamp #1: fredag 2002-02-15 16:49:58 (local time)

Time stamp #2: fredag 2002-02-15 16:49:58 (local time)

block_6058_type_c.bin

FC 6058

Function family: IEC

Function name: DT_DATE
Created in: STL (STatement List)
Time stamp #1: fredag 2002-02-15 16:49:51 (local time)
Time stamp #2: fredag 2002-02-15 16:49:51 (local time)

block_6059_type_c.bin
FC 6059
Function family: IEC
Function name: NA_ME
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 18:55:44 (local time)
Time stamp #2: måndag 2007-09-24 18:55:44 (local time)

block_6060_type_c.bin
FC 6060
Function family: IEC
Function name: CALC
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:30 (local time)
Time stamp #2: måndag 2007-09-24 18:57:30 (local time)

block_6061_type_c.bin
FC 6061
Function family: IEC
Function name: DONE
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 18:56:48 (local time)
Time stamp #2: måndag 2007-09-24 18:56:48 (local time)

block_6062_type_c.bin

FC 6062
Function family: IEC
Function name: INIT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:42 (local time)
Time stamp #2: måndag 2007-09-24 18:57:42 (local time)

block_6063_type_c.bin
FC 6063
Function family: IEC
Function name: IO_ST
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:38 (local time)
Time stamp #2: måndag 2007-09-24 18:57:38 (local time)

block_6064_type_c.bin
FC 6064
Function family: IEC
Function name: RD_ST
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:36 (local time)
Time stamp #2: måndag 2007-09-24 18:57:36 (local time)

block_6065_type_c.bin
FC 6065
Function family: IEC
Function name: DUMP_DT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:32 (local time)
Time stamp #2: måndag 2007-09-24 18:57:32 (local time)

block_6066_type_c.bin
FC 6066
Function family: IEC
Function name: MOD_NM
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:34 (local time)
Time stamp #2: måndag 2007-09-24 18:57:34 (local time)

block_6067_type_c.bin
FC 6067
Function family: IEC
Function name: MAIN
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 18:56:05 (local time)
Time stamp #2: måndag 2007-09-24 18:56:05 (local time)

block_6068_type_c.bin
FC 6068
Function family: IEC
Function name: GET_ST
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 18:55:22 (local time)
Time stamp #2: måndag 2007-09-24 18:55:22 (local time)

block_6069_type_c.bin
FC 6069
Function family: IEC
Function name: RD_PH
Created in: STL (STatement List)

Time stamp #1: måndag 2007-09-24 18:56:27 (local time)
Time stamp #2: måndag 2007-09-24 18:56:27 (local time)

block_6070_type_c.bin
FC 6070
Function family: IEC
Function name: AFL_OP
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:28 (local time)
Time stamp #2: måndag 2007-09-24 18:57:28 (local time)

block_6071_type_c.bin
FC 6071
Function family: IEC
Function name: AVERGE
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:26 (local time)
Time stamp #2: måndag 2007-09-24 18:57:26 (local time)

block_6072_type_c.bin
FC 6072
Function family: IEC
Function name: PRM_DT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:24 (local time)
Time stamp #2: måndag 2007-09-24 18:57:24 (local time)

block_6073_type_c.bin
FC 6073
Function family: IEC

Function name: IS_OP
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:19 (local time)
Time stamp #2: måndag 2007-09-24 18:57:19 (local time)

block_6074_type_c.bin
FC 6074
Function family: IEC
Function name: UP_STRNG
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:17 (local time)
Time stamp #2: måndag 2007-09-24 18:57:17 (local time)

block_6075_type_c.bin
FC 6075
Function family: IEC
Function name: LGC_OP
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:40 (local time)
Time stamp #2: måndag 2007-09-24 18:57:40 (local time)

block_6076_type_c.bin
FC 6076
Function family: IEC
Function name: SAV_MOVB
Created in: STL (STatement List)
Time stamp #1: måndag 2007-09-24 18:54:39 (local time)
Time stamp #2: måndag 2007-09-24 18:54:39 (local time)

block_6077_type_c.bin

FC 6077
Function family: IEC
Function name: RND_OP
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:15 (local time)
Time stamp #2: måndag 2007-09-24 18:57:15 (local time)

block_6078_type_c.bin
FC 6078
Function family: IEC
Function name: SB_DT_NM
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:12 (local time)
Time stamp #2: måndag 2007-09-24 18:57:12 (local time)

block_6079_type_c.bin
FC 6079
Function family: IEC
Function name: CO_DAT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:22 (local time)
Time stamp #2: måndag 2007-09-24 18:57:22 (local time)

block_6080_type_c.bin
FC 6080
Function family: IEC
Function name: ROD_NM
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:20 (local time)
Time stamp #2: måndag 2007-09-24 18:57:20 (local time)

block_6081_type_c.bin
FC 6081
Function family: IEC
Function name: NR_DT
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:11 (local time)
Time stamp #2: måndag 2007-09-24 18:57:11 (local time)

block_6082_type_c.bin
FC 6082
Function family: IEC
Function name: AD_OP
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:44 (local time)
Time stamp #2: måndag 2007-09-24 18:57:44 (local time)

block_6083_type_c.bin
FC 6083
Function family: IEC
Function name: TMR_DB
Created in: SCL (Structured Control Language)
Time stamp #1: måndag 2007-09-24 18:57:46 (local time)
Time stamp #2: måndag 2007-09-24 18:57:46 (local time)

block_6084_type_c.bin
FC 6084
Function family: IEC
Function name: RD_SK
Created in: STL (STatement List)

Time stamp #1: måndag 2007-09-24 18:55:00 (local time)
Time stamp #2: måndag 2007-09-24 18:55:00 (local time)

block_8062_type_a.bin
DB 8062
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 18:53:29 (local time)
Time stamp #2: måndag 2007-09-24 18:53:29 (local time)

block_8063_type_a.bin
DB 8063
Function family:
Function name:
Created in: DB (Data Block)
Time stamp #1: måndag 2007-09-24 18:53:27 (local time)
Time stamp #2: måndag 2007-09-24 18:53:27 (local time)

block_80_type_8.bin
OB 80
Function family:
Function name:
Created in: FBD (Function Block Diagram)
Time stamp #1: torsdag 2007-02-08 15:04:47 (local time)
Time stamp #2: fredag 2002-02-15 16:51:13 (local time)