MEMORIE

DELL'ACCADEMIA DELLE SCIENZE DI TORINO

2

SERIE VI

2023

ESTRATTI

Acc. Sc. Torino
Memorie 2 (2023), 5-23, 1 fig.

Cybersecurity per *Internet of Things* e infrastrutture critiche

Edoardo Calia[**], Marco Mezzalama[*] e Andrea Vesco[**]

Conferenza tenuta dal Socio nazionale Marco Mezzalama
nell'adunanza del 13 aprile 2022, testo ricevuto il 19 maggio 2023,
approvato come memoria nell'adunanza del 7 giugno 2023,
online dal 3 ottobre 2023

Abstract. *This paper addresses the issue of cybersecurity of smart objects (generally referred to as IoTs), with a focus on issues related to industrial facilities monitored and controlled by such IoTs. After discussing some real-world cases where serious problems have occurred due to unauthorized, if not criminal, intrusions into industrial facilities, the paper reviews techniques that can be adopted to prevent or at least reduce the possibility of intrusion and/or tampering into IoT systems for monitoring critical infrastructure and complex business systems.*

Keywords: cybersecurity, identity, integrity, IoT, security of industrial facilities.

Riassunto. *Il presente articolo intende affrontare il tema della sicurezza informatica degli oggetti intelligenti (in generale indicati come IoT), con particolare riguardo alle problematiche legate agli impianti industriali da questi ultimi monitorati e controllati. Dopo la discussione di una serie di casi in cui si sono verificate grandi problematicità a causa dell'intrusione non autorizzata, se non criminale, a impianti industriali, l'articolo passa in rassegna le tecniche che possono essere adottare per eliminare o per lo meno ridurre la possibilità di intrusione e/o manomissione su sistemi IoT per il controllo di infrastrutture critiche e sistemi aziendali complessi.*

Parole chiave: identità, integrità, IoT, sicurezza delle infrastrutture industriali, sicurezza informatica.

[*] Politecnico di Torino, Accademia delle Scienze, Fondazione Links; marco.mezzalama@polito.it
[**] Fondazione Links; edoardo.calia@linksfoundation.com; andrea.vesco@linksfoundation.com

Introduzione

L'evoluzione tecnologica nei settori dell'elettronica e delle telecomunicazioni ha favorito nei due decenni passati la diffusione di oggetti dotati di capacità di calcolo e di trasmissione dati: si tratta di fatto di piccoli elaboratori, simili – dal punto di vista dell'architettura interna – ai computer che siamo abituati ad utilizzare quotidianamente, anche se normalmente privi dei classici strumenti di interazione con gli utenti quali monitor, mouse, tastiera.

Questi oggetti *intelligenti* e *connessi* risultano molto utili nella gestione delle attività quotidiane: da quelle più semplici e personali (si pensi agli smart watch, alle videocamere di sorveglianza domestica, ai cosiddetti *home assistant*, e ovviamente agli smartphone) fino a quelle professionali – e in alcuni casi critiche per ragioni di sicurezza – quali il controllo di impianti, di centrali elettriche, infrastrutture stradali, il controllo accessi. La dotazione di oggetti smart consente infatti di poter *osservare*, e in alcuni casi *controllare*, quanto avviene nelle nostre case, nei nostri uffici, nelle nostre aziende, nelle nostre città. È il mondo della cosiddetta *Internet of Things (IoT)*, la *rete degli oggetti*.

Questi dispositivi sono al centro della silenziosa ma inesorabile trasformazione digitale del mondo nel quale viviamo in un mondo intelligente: una rivoluzione destinata ad avere un impatto sulla nostra vita analogo a quelli portati dalla introduzione dell'energia elettrica, dei motori elettrici e termici, e più recentemente della rete Internet.

Il mondo che stiamo rapidamente costruendo è quindi un mondo nel quale, grazie alle tecnologie di calcolo e di connessione, è possibile collegarsi da qualsiasi luogo ad uno o più di questi oggetti connessi, per controllare, ad esempio, la temperatura del nostro alloggio ed eventualmente accendere il riscaldamento prima di arrivare a casa oppure per regolare le paratoie di una diga per aumentare o diminuire il flusso di scarico di un bacino idrico.

Gli esempi precedenti di applicazioni rese possibili da questi sofisticati oggetti intelligenti evidenziano un'altra delle loro caratteristiche peculiari che li differenziano dagli elaboratori tradizionali: la stretta connessione con il mondo fisico, che si manifesta nella possibilità di attivare sistemi robotici e meccanici completando azioni anche al di fuori del mondo digitale. Questa caratteristica conferisce agli impianti e ai sistemi dotati di intelligenza e connettività grazie alla presenza di questi *smart object* un nuovo nome: quello di *Sistemi Cyber-Fisici*, o *Cyber-Physical Systems*.

In questo articolo vogliamo evidenziare alcune possibili criticità che derivano dalla presenza di oggetti intelligenti all'interno della architettura di infrastrutture importanti, e in particolare ai rischi – e alle relative tecniche di mitigazione – che questa nuova configurazione comporta dal punto di vista

della sicurezza informatica o *cyber security*. Come già accennato, gli oggetti connessi posseggono infatti le due caratteristiche fondamentali che consentono – ad esempio – ad un virus informatico o malware di operare e propagarsi: una CPU che ne esegua il software, e una rete per diffonderlo, oltre che inviare o ricevere informazioni da altri oggetti o elaboratori.

1. Infrastrutture critiche *smart*: un nuovo target per gli attacchi informatici

Chi ha vissuto la storia dell'informatica negli ultimi 40 anni ricorda certamente l'entusiasmo con il quale intorno all'inizio degli anni '80 è stata salutata la possibilità di interconnettere tra loro sistemi di elaborazione: una rivoluzione che presto portò alla possibilità di spostare dati e informazioni da una parte all'altra del mondo in pochi secondi, resa possibile dalle stesse tecnologie di comunicazione sulle quali si basa ancora oggi la rete Internet. A quell'entusiasmo è rapidamente seguita la constatazione che gli usi della rete potessero anche andare oltre quelli progettati e desiderati: a poca distanza dalla introduzione delle reti di comunicazione digitali abbiamo assistito alla nascita delle azioni mirate ad accedere illegalmente agli elaboratori (ora raggiungibili facilmente grazie alla connettività globale) per sottrarre informazioni riservate e rilevanti dal punto di vista industriale, politico o militare.

Tali azioni, delle quali le prime tracce risalgono alle metà degli anni '80, hanno trovato terreno fertile nelle prime implementazioni delle reti digitali, progettate senza tecniche native di sicurezza informatica. E a dire il vero ancora oggi la rete Internet non è *intrinsecamente* sicura: essa trasporta i dati nella forma in cui sono generati, lasciando agli elaboratori che si scambiano le informazioni il compito di applicare alla fonte tecniche di cifratura, autenticazione degli utenti, autorizzazione all'accesso.

La diffusione di oggetti connessi come parte integrante dei sistemi di monitoraggio e controllo di impianti e infrastrutture critiche espone questi sistemi a minacce del tutto simili a quelle che nei 40 anni passati hanno interessato i sistemi informativi di tutto il mondo. Ma se finora la intrusione in un sistema digitale poteva essere finalizzato alla sottrazione di dati o di denaro, è facile immaginare come analoghi attacchi a sistemi cyber-fisici possano avere conseguenze ben più drammatiche, mettendo a rischio la vita di migliaia di persone (ad esempio aumentando la immissione in un acquedotto di sostanze chimiche che in piccole dosi sono necessarie per la purificazione dell'acqua) o l'operatività di intere città (ad esempio attaccando il sistema di controllo di una rete elettrica o del gas).

Nei 20 anni passati (ovvero poco dopo la introduzione sul mercato di oggetti e sensori connessi in rete) si sono verificati in diverse parti del mondo attacchi informatici a infrastrutture critiche che hanno rivelato come tutto ciò sia tutt'altro che una remota ipotesi: come meglio dettagliato negli esempi riportati nelle sezioni seguenti, gli oggetti connessi (che oggi sono la comunità di utenti più numerosa della rete Internet[1]) sono diventati obiettivo di azioni in tutto assimilabili ad attacchi terroristici su grande scala. E il fenomeno sta crescendo rapidamente: gli attacchi di tipo cyber condotti verso impianti industriali o infrastrutture critiche sono aumentati numericamente del 200% all'anno nell'ultimo biennio. L'impatto economico dei *cyber crimini* (non limitati a quelli che hanno come obiettivo le infrastrutture) è di circa 6,000 miliardi di dollari, con un costo medio di ciascuna violazione pari a 3,86 milioni di dollari (dati 2020).

La complessità nel contrasto ai cyber attacchi è sempre crescente in quanto le tecniche utilizzate sono sempre più sofisticate, e gli hacker celano l'attacco dietro situazioni che apparentemente sono del tutto normali: il 90% degli attacchi ha inizio perché un utente apre un allegato o un link contenuto in un messaggio email apparentemente lecito e innocuo; il 38% dei virus e dei malware vengono installati sugli elaboratori degli utenti nascosti sotto forma di *macro* in documenti di testo (Word).

Gli attacchi a impianti e infrastrutture presentano tuttavia aspetti di complessità che li rendono più difficili da mettere a punto rispetto agli attacchi che riguardano in modo esclusivo il mondo digitale e gli elaboratori tradizionali: l'architettura degli elaboratori e le modalità della loro connessione in rete sono infatti relativamente standard, e quindi la propagazione di un virus o le tecniche di penetrazione di un server connesso a Internet sono valide su scala quasi universale. Viceversa gli attacchi che hanno come target sistemi cyber-fisici devono fare i conti con il fatto che per questi ultimi non esiste in genere uno standard di realizzazione o implementazione. Spesso quindi l'attacco cyber a un impianto richiede un sopralluogo sul posto per verificare quali macchinari sono presenti e qual è la tipologia di dispositivi di controllo utilizzata. Le motivazioni che spingono gli attaccanti a scegliere questo tipo di obiettivi vanno generalmente oltre la semplice sottrazione di dati o danneggiamento di un sistema informatico (possono essere ragioni politiche, sociali oltre che meramente economiche), e spesso giustificano – dal punto di vista dell'aggressore – una attività organizzativa costosa e di lungo periodo prima di sferrare l'attacco.

---

[1] Il numero degli oggetti connessi alla fine del 2022 è stimato in 14.5 miliardi, di gran lunga superiore al numero di utenti umani (https://iot-analytics.com).

Stuxnet: attacco ai laboratori nucleari di Natanz (Iran)

Il caso del virus Stuxnet è il primo esempio di attacco cyber diretto ad un impianto industriale. Stuxnet è un virus che nel 2010 è stato utilizzato per mettere fuori uso alcune centrifughe utilizzate per l'arricchimento dell'uranio in una centrale nella città di Natanz (Iran).

Il software, estremamente sofisticato, si comportava come un *worm*, ovvero era in grado di propagarsi sui normali elaboratori del centro di controllo della centrale. Da questi eseguiva la installazione di due moduli software esclusivamente su alcuni dispositivi (chiamati PLC) usati per il controllo dei macchinari *target*: il primo modulo andava a sostituire il software di controllo delle turbine, e il secondo progettato per inviare alla stazione di controllo indicazioni di funzionamento normale, nascondendo così l'attacco. Prima che gli addetti all'impianto si rendessero conto di quanto stava succedendo, numerose delle centrifughe dell'impianto sono state distrutte dal virus, che agiva (causando un grave impatto nel *mondo fisico*) alterando in modo irregolare il numero di giri del motore dei macchinari.

Nel caso di Stuxnet la responsabilità della propagazione iniziale del *worm* non è della rete: l'impianto era infatti protetto e completamente isolato, e quindi non accessibile dal mondo esterno utilizzando la rete Internet. Si ritiene che il software malevolo sia stato introdotto nell'impianto utilizzando una chiavetta USB in dotazione al personale che eseguiva la manutenzione o aggiornamento software dei dispositivi.

Il livello di sofisticazione dell'attacco lascia pensare che esso sia stato messo a punto da organizzazioni governative di stati stranieri ostili all'Iran. La notizia sull'attacco e le informazioni sul software Stuxnet sono state diffuse per errore quando il virus è inavvertitamente uscito dalla centrale a bordo di un laptop che era stato colpito, e che ha poi propagato il worm all'esterno dell'area confinata alla quale era stato destinato. All'esterno della centrale il virus non ha fatto danni in quanto era stato progettato per operare in modo specifico nel contesto di quell'impianto, dimostrando tra l'altro che chi lo ha progettato conosceva bene l'ambiente informatico utilizzato.

Il primo attacco ad una centrale elettrica (Ucraina, dicembre 2015)

Nel 2015 è stato registrato in Ucraina il primo attacco ad una centrale elettrica, che ha lasciato senza alimentazione elettrica più di 200,000 persone per diverse ore. L'attacco, verificatosi nel contesto storico della guerra tra Russia e Ucraina iniziata l'anno precedente, è stato attribuito a hacker che hanno agito via Internet dalla Federazione Russa.

L'attacco, pianificato durante il periodo natalizio nel quale la centrale aveva in sede una forza lavoro molto ridotta, è stato organizzato in diverse fasi, alcune delle quali sono state condotte in parallelo:

- Attacco e penetrazione all'interno della rete dell'azienda elettrica, utilizzando la tecnica *email phishing*[2] per propagare il software malevolo, denominato *BlackEnergy*
- Acquisizione del controllo (tastiera, mouse, monitor) di workstation utilizzate per la gestione del sistema. Conseguente spegnimento manuale di diverse sottostazioni
- Parallelamente, messa fuori uso di diversi elementi della infrastruttura IT dell'azienda (UPS, Modem etc)
- Distruzione di file utili per la riconfigurazione dei sistemi
- Attacco di tipo DOS (Denial-Of-Service[3]) verso il call center della società, impedendo la comunicazione da e verso gli utenti vittime della interruzione di energia elettrica

È degno di nota il fatto che l'anno successivo, sempre nel periodo delle festività del Natale, l'Ucraina fu vittima di un secondo attacco. A differenza dell'anno precedente, il virus utilizzato (chiamato *Industroyer*) questa volta era più evoluto e in grado di eseguire la procedura di spegnimento delle sottostazioni in modo completamente automatico (senza cioè l'intervento manuale – ancorché da remoto – dell'attaccante).

Ransomware: l'attacco a Colonial Pipeline (Maggio 2021)

La Colonial Pipeline è un oleodotto che si estende per 9000 Km dal Texas al New Jersey, portando carburante in molti stati della costa Est degli Stati Uniti. Nel 2021 fu vittima del più grande *cyber attacco* ad una infrastruttura critica negli USA.

L'attacco in questo caso non è stato diretto ai dispositivi installati lungo la infrastruttura di trasporto, ma indirettamente passando dalla rete dell'azienda dove venne eseguita la intrusione in alcuni server dai quali venne sottratta una grande quantità di dati in due ore. Terminata la sottrazione dei dati la rete aziendale fu contaminata con un *ransomware*, un particolare tipo di virus che

---

[2] La tecnica *email fishing* consiste nell'inviare ad un utente messaggi di email in tutto simili a messaggi leciti (falsificando sia il mittente sia il testo del messaggio), nei quali viene chiesto di svolgere azioni apparentemente necessarie ma che in realtà sono studiate per sottrarre credenziali o altre informazioni utili per compiere atti criminosi (es. accesso a siti di *home banking* o installazione di software malevolo sul computer dell'utente).

[3] Gli attacchi di tipo *Denial of Service* vengono perpetrati generando un numero di richieste verso un servizio tanto elevato da mandare in crisi i sistemi informatici che lo erogano, rendendo di fatto il servizio non utilizzabile dagli utenti per tutta la durata dell'attacco.

ha l'effetto di cifrare – rendendoli illeggibili – i dati sugli elaboratori attaccati.

La società, accortasi della presenza del virus, decise cautelativamente di limitarne la propagazione spegnendo tutti gli impianti (inclusi quelli di pompaggio), e sospendendo quindi la fornitura di carburante lungo tutto l'oleodotto.

L'attaccante, un gruppo che si fa chiamare DarkSide, chiese un riscatto (da cui il nome di *ransomware* attribuito a questo tipo di attacchi) di 75 bitcoin (pari ad oggi a circa 1,25 milioni di €, e al tempo dell'attacco a circa 3 milioni di €), che Colonial Pipeline pagò per ottenere il ripristino dei dati cifrati e recuperare il controllo dei propri sistemi. Il disservizio ebbe una durata totale di 6 giorni (una parte significativa del riscatto venne successivamente recuperato tracciando le transazioni eseguite per lo spostamento dei Bitcoin).

Il canale utilizzato per l'attacco fu in questo caso la VPN aziendale[4]: la password di uno dei collaboratori di Colonial Pipeline era stata catturata in un diverso attacco non collegato all'azienda (l'utente utilizzava la stessa password per accedere a diversi servizi o applicazioni anche ad uso personale).

2. Cybersecurity degli oggetti connessi

Se oggi la cyber security per i sistemi di informazione "classici" (server, portali web, desktop, laptop) è un tema al quale sono dedicati una elevata attenzione e investimenti significativi, lo stesso non si può dire per la sicurezza informatica degli oggetti connessi: chiunque abbia acquistato negli anni passati uno di questi dispositivi *smart* (per la gestione della casa ad esempio) avrà certamente notato che la configurazione della sicurezza o non è disponibile, oppure è limitata alla impostazione di semplici password. E questi oggetti non sono più *innocui* come i primi dispositivi di tipo IoT: hanno una elevata capacità di calcolo (e sono quindi in grado di eseguire applicazioni software complesse) grazie ai progressi delle tecnologie digitali (processori sempre più piccoli e veloci, pur necessitando di poca energia). E si trovano connessi in un contesto nel quale le grandi dorsali di rete hanno una capacità praticamente illimitata: basti pensare che le moderne fibre ottiche sono in grado di trasportare 1 Petabit (ovvero un milione di miliardi di bit) al secondo ciascuna.

---

[4] La tecnologia VPN (*Virtual Private Network*) consente agli utenti di virtualizzare (rendendola disponibile da remoto) la propria rete aziendale, consentendo cioè loro di utilizzare i servizi e i sistemi della propria azienda come se si trovassero nei propri uffici. Per evidenti ragioni di sicurezza il servizio VPN è attivabile solo avendo opportune credenziali.

Gli oggetti all'attacco: la tecnica DDOS (Distributed Denial-Of-Service)

I rischi *cyber* connessi alla presenza nel mondo di oggetti in grado di elaborare e comunicare non sono solo gli attacchi che hanno gli stessi oggetti come obiettivo finale: altrettanto critiche sono nuove tipologie di attacco che utilizzano gli oggetti come armi digitali per mettere fuori uso altri obiettivi (server o servizi).

Tra questi è particolarmente rilevante (e purtroppo diffusa) la tecnica detta Distributed Denial of Service, il cui nome deriva dal fatto che l'attacco non parte da un solo elaboratore, ma richiede una preventiva organizzazione di una sorta di "esercito" (*IoT army*) di centinaia o migliaia di dispositivi istruiti in modo che, al momento predefinito, si attivino in modo coordinato per condurre l'azione criminosa (in questo caso la interruzione, seppur temporanea, di un servizio[5]).

L'uso di una tecnica così complessa, che richiede una attenta e lunga fase di preparazione, è giustificata da alcuni fattori principali:

- La potenza di calcolo degli elaboratori (e quindi dei *server* utilizzati per erogare servizi) è finora cresciuta esponenzialmente nel tempo: oggi è possibile mettere in crisi un servizio solo se si riesce a "bombardare" i sistemi corrispondenti con una quantità di richieste estremamente elevata.
- Come fatto notare nella sezione precedente, le dorsali della rete globale (alla quale sono connessi gli elaboratori che erogano i principali servizi) sono in grado di *trasportare* la mole di dati necessaria per un attacco di DOS. Viceversa la connettività e la potenza di calcolo a disposizione di un comune elaboratore installato nella periferia della rete non sono sufficienti a *generare* quel traffico. È quindi necessario ricorrere a più punti di generazione di traffico localizzati in periferia, e convogliare quel traffico di rete verso il server *target* dell'attacco.
- La presenza di un elevato numero di oggetti connessi a Internet e la loro ampia e capillare distribuzione nella periferia della rete costituisce lo scenario ideale per configurare e mettere in atto gli attacchi di *DOS distribuito*

Le fasi nelle quali si articola un moderno attacco DDOS sono quindi le seguenti:

- L'attaccante esegue una ampia scansione della rete internet alla

[5] Si noti che gli attacchi di tipo DOS o DDOS non arrecano danni permanenti ai sistemi presi di mira: l'obiettivo dell'attacco in questi casi è causare la non disponibilità del servizio target per un periodo di tempo che dipende dalle capacità e modalità di reazione dell'attaccato.

ricerca di oggetti connessi che presentino vulnerabilità di accesso e funzionalità utili per l'attacco (assenza di password, password semplici o pubblicamente note, possibilità di installazione ed esecuzione di software, etc)

- Su ciascuno di questi oggetti viene installato un piccolo programma (applicazione SW) che non fa altro che inviare una sequenza di richieste al server che si desidera rendere indisponibile (DOS): un esempio semplice di interrogazione potrebbe essere la richiesta di una specifica pagina web. Questa applicazione viene programmata in modo identico su tutti i dispositivi che fanno parte della *IoT army* per rimanere inizialmente inattiva (anche per un tempo lungo), entrando in azione ad una specifica data e ora: per evidenti ragioni questa configurazione viene spesso indicata come una *bomba a tempo* (*time bomb*).

A questo punto l'attacco è pronto (si noti che fino a questo punto, salvo errori dell'attaccante, nessuno ha avuto modo di accorgersi di quanto è avvenuto), e al momento prestabilito tutti gli oggetti selezionati e programmati inizieranno a inviare (tutti nello stesso istante) il maggior numero di richieste possibile verso il server target. Il traffico generato sarà convogliato sulle dorsali verso la vittima, che riceverà uno *tsunami* di richieste che – se l'attacco è stato progettato in modo adeguato – lo metterà fuori uso.

Esempio di DDOS: l'attacco a Dyn (ottobre 2016)

Dyn era una società specializzata in servizi di cybersecurity e nella osservazione, controllo e ottimizzazione di infrastrutture online. Tra i servizi che offriva figurava anche la gestione del servizio DNS (Domain Name System[6]) per ampie aree geografiche degli Stati Uniti.

Il servizio DNS è fondamentale per poter raggiungere qualsiasi sito o elaboratore connesso a Internet. Una eventuale indisponibilità di quel servizio mette in ginocchio l'intera connettività da e verso la rete internet per gli utenti delle aree geografiche interessate.

Nell'Ottobre 2016 i server della società Dyn sono stati obiettivo di tre attacchi di tipo DDOS consecutivi, isolando di fatto dalla rete milioni di utenti

[6] Il servizio DNS è uno dei servizi basilari della rete Internet, e si occupa della traduzione da indirizzi mnemonici – come quelli che indichiamo come URL per raggiungere siti web – nei corrispondenti indirizzi numerici (molto più difficili da ricordare) che sono utilizzati da elaboratori, router e altri apparati per le comunicazioni sulla rete Internet. Ad esempio, ogni volta che consultiamo un sito Internet il nostro elaboratore esegue come prima operazione una richiesta al servizio DNS per ottenere l'indirizzo numerico corrispondente al sito che ci interessa visitare.

per diverse ore, e alcuni siti molto utilizzati come Amazon, Twitter, Spotify, Netflix, Reddit risultavano irraggiungibili.

L'attacco è stato condotto inviando a Dyn richieste contemporanee di traduzione da nome a indirizzo IP (il servizio principale del DNS) da decine di milioni di falsi utenti, utilizzando decine di migliaia di oggetti connessi.

Il software (malware) utilizzato, noto con il nome di *Mirai botnet*, in preparazione per l'attacco era stato installato su un elevato numero (tra 50,000 e 100,000) di dispositivi come webcam, router domestici e *baby monitor* in tutto il mondo. Come in altri casi, Mirai era costituito da due parti, una dedicata alla propagazione (il *worm*) e l'altra alla esecuzione dell'attacco (*Command and Control*).

Una curiosità relativa a Mirai Botnet è che il suo codice è stato reso pubblico, ed è stato utilizzato da molti per mettere a punto attacchi simili a quello diretto a Dyn. Tra questi l'esempio più rilevante (verificatosi nel novembre 2016, solo un mese dopo l'attacco ai server di Dyn) ha interessato 900,000 router di Deutsche Telecom, sui quali una versione modificata di Mirai è stata installata sfruttando una vulnerabilità del sistema di accesso per la gestione remota di quegli apparati.

Tecniche di difesa: come proteggere gli oggetti connessi?

Gli esempi di attacchi perpetrati utilizzando oggetti connessi (come *obiettivo ultimo* oppure come *strumento di esecuzione* del crimine) hanno un fattore in comune: la violazione degli oggetti – sfruttando vulnerabilità dovute alla loro configurazione SW o a problemi di progettazione – per installarvi componenti del malware utilizzato. Il problema sarebbe risolto se esistesse un modo per rendere complessa o impossibile la installazione di software non certificato su questi dispositivi. Ulteriori garanzie di cybersecurity si otterrebbero inserendo nella architettura delle infrastrutture che fanno uso di IoT sistemi sicuri e robusti preposti a verificare che gli oggetti connessi siano quelli autorizzati, e che il software che hanno a bordo non sia stato modificato.

La concorrenza di queste soluzioni contribuisce a creare un ambiente di esecuzione del software sicuro e affidabile, che tecnicamente viene chiamato Trusted Environment. Tra gli elementi basilari che lo compongono figurano ingredienti e tecnologie quali la *gestione della identità digitale* degli oggetti, le *tecniche di Embedded Trusted Computing* e quelle di *decentralizzazione e ridondanza* tipiche delle architetture come *BlockChain* e – più in generale – *Distributed Ledger (DLT)*.

Identità digitale degli oggetti connessi

Il tema della gestione della identità digitale ha una lunga storia nel mondo dei sistemi informativi e della rete Internet, e trova una delle possibili soluzioni nei cosiddetti certificati digitali. Costruito sulla base di un particolare sistema crittografico detto PKI (*Public Key Infrastructure*), il certificato digitale è "*un documento elettronico che attesta la associazione univoca tra una chiave pubblica e l'identità di un soggetto (una persona, una organizzazione, un hardware, un servizio, un software, un file, etc)"* (Wikipedia). La caratteristica delle architetture PKI di prevedere l'esistenza di una chiave pubblica e una chiave privata conferisce a questa famiglia di soluzioni anche il nome di crittografia asimmetrica[7].

Tra le informazioni presenti nel certificato figura la firma (digitale) della autorità che ha verificato la correttezza delle informazioni e la identità del soggetto interessato.

Una trattazione dettagliata della architettura PKI esula dagli scopi di questo articolo: è sufficiente qui sottolineare come un utente che sia in possesso della chiave pubblica associata ad un determinato certificato può avere la certezza della identità di chi possiede il certificato digitale (garantita dall'autorità che ha emesso il certificato). Tra le azioni che possono essere svolte utilizzando la chiave pubblica figurano:

- La verifica che un determinato documento digitale (o un file: una immagine, *una porzione di SW*) sia autentico e non sia stato modificato in alcun modo, purché rechi la firma di chi possiede il certificato corrispondente a quella chiave pubblica
- Lo scambio di dati in formato cifrato, e quindi non intercettabile da terzi

È quindi intuibile come un oggetto connesso possa essere reso più sicuro assegnandogli un certificato digitale in modo simile a quanto si fa con gli utenti umani (sulle *smart card* che utilizziamo per apporre la firma digitale ai documenti in formato elettronico – che assumono in quel modo validità legale – è di fatto memorizzato uno di questi certificati associato in modo inequivocabile alla nostra identità). Rimane tuttavia aperto il tema di quale

[7] Il principio matematico sul quale si basano le tecniche di *crittografia asimmetrica* prevede la generazione di due chiavi, una delle quali – scelta indifferentemente – può essere utilizzata per cifrare un documento, e l'altra per decifrarlo. I servizi abilitati dalle architetture PKI (verifica della firma digitale, verifiche di integrità del documento firmato, funzione di *non ripudio* etc) richiedono che a una di queste chiavi venga attribuito il ruolo di *chiave pubblica* (ovvero che può essere diffusa pubblicamente), e all'altra il ruolo di *chiave privata* (la quale non deve essere diffusa, ma custodita e utilizzata solo dal suo titolare).

sia l'autorità incaricata di assegnare l'identità ad uno specifico dispositivo (occorrerebbe una sorta di *Anagrafe degli oggetti connessi*).

Se gli oggetti avessero tutti un certificato digitale, sarebbe possibile *autenticare* ciascuno di essi, verificare cioè (utilizzando la sua chiave pubblica, che per definizione sarebbe nota a chiunque) se un determinato oggetto è stato sostituito come parte preparatoria di un attacco.

La *autenticazione* di un utente è il primo passo per stabilire con esso un rapporto di fiducia, che viene completato eseguendo una fase di *autorizzazione* ad accedere alle risorse alle quali ha diritto.

La flessibilità tipica delle tecnologie digitali consente una gestione della identità molto più sofisticata di quanto avviene nel mondo "analogico": in numerose occasioni (accesso a un locale, un ufficio, oppure situazioni nelle quali occorre dimostrare la residenza, l'età, o altro) siamo abituati a farci riconoscere esibendo un documento che contiene molte informazioni relative alla nostra persona, gran parte delle quali non sono necessarie per ottenere il servizio di interesse. Sarebbe molto più coerente (e più rispettoso della nostra privacy) avere documenti diversi, selettivi per singole specifiche occasioni: un documento per dimostrare solo la nostra età, uno per dimostrare solo la nostra la residenza, un altro per dimostrare che siamo autorizzati alla guida di un certo veicolo, un altro per dimostrare il possesso di un immobile etc. La gestione della identità digitale rende possibile tutto questo, conferendo ad un utente (sia esso un essere umano o un oggetto) un insieme ampio di credenziali verificabili singolarmente e semplici da utilizzare.

La critica principale ai sistemi centralizzati per la gestione dell'identità risiede nella presenza di una autorità centrale che rilascia le credenziali e permette di verificare l'identità di un oggetto. In un mondo informatico sempre più basato sul concetto di "distribuzione" è possibile superare questo paradigma adottando nuove soluzioni che vanno sotto il nome di *Self-Sovereign Identity* (SSI).

La Self-Sovereign Identity è un nuovo modello di identità digitale per utenti di Internet, e già si presenta come una delle tendenze più importanti dei prossimi decenni. Il principio fondante della SSI è quello di spostare il controllo dell'identità digitale dai "fornitori di identità" terzi direttamente agli individui: questo significa che il controllo delle informazioni relative alla propria identità viene affidato all'utente. Il framework SSI sfrutta il paradigma dell'identità decentralizzata e si basa su due importanti standard del World Wide Web Consortium (W3C): i Decentralized IDentifiers (DID) e le Verifiable Credentials (VC). Il riferimento a questi standard W3C rende il framework SSI altamente interoperabile e portabile.

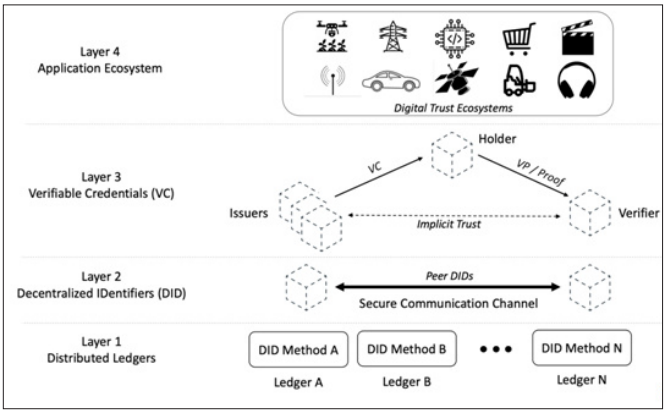

Fig. 1. Lo stack completo della SSI.

Lo stack completo della SSI è rappresentato nella Figura 1.

Il **livello 1** si occupa della *memorizzazione delle informazioni di identità*, ed è costituito da una architettura di tipo *Distributed Ledger* (DLT) che funge da *radice della fiducia* o *Root-of-Trust* (RoT) dell'intera struttura. La DLT funziona come un archivio distribuito nel quale la fiducia e la immutabilità sono garantite dal protocollo di consenso della DLT. I protocolli di consenso sono un tema ampiamente dibattuto e studiato nella letteratura specialistica, ma limitando la classificazione alle diverse tipologie di registro o libro mastro (*ledger*) utilizzato è possibile identificarne due classi principali: il consenso utilizzato nelle *blockchain* (ad esempio Bitcoin, Ethereum) e quello utilizzato nei ledger di tipo *grafo aciclico diretto* o *DAG* (come quello adottato da IOTA, e che prende il nome di *Tangle*).

Un IDentificatore decentralizzato è un nuovo tipo di identità, unica a livello globale, progettata per essere assegnata ad un soggetto (ad esempio, esseri umani e cose) il quale ne mantiene il pieno controllo, in conformità con il modello di auto-sovranità. Le DID sono rappresentate con il formato URI (*Uniform Resource Identifier*), e associano un soggetto (il *DID subject*) a un documento (*DID document*) che contiene le informazioni necessarie per interagire in modo affidabile con quel soggetto. Il livello 1 comprende anche il *DID method*, una porzione di codice software che abilita lo scambio di informazioni con il *ledger* specifico. Un *DID method* deve fornire le primitive per

- *creare l'identità* (cioè generare due coppie di chiavi e il *DID document*, e memorizzarlo nel *ledger* all'indirizzo specificato dal DID)

- risolvere un DID (cioè recuperare un *DID document* dall'indirizzo del ledger a cui punta il DID e verificare la correttezza del formato del *DID document*)
- aggiornare il DID (cioè generare un nuovo DID e un nuovo *DID document*)
- revocare un DID (cioè fornire un'evidenza nel ledger del fatto che un DID e il relativo *DID document* siano stati revocati dal controllore).

L'implementazione del *DID method* è specifica per il ledger scelto, e rende i livelli superiori indipendenti da esso.

Il **livello 2** fa uso delle DID e dei *DID Document* per autenticare un partner di una comunicazione e *stabilire con esso un canale di scambio sicuro* (garantendo cioè *mutua autenticazione*, *riservatezza* e *integrità*) secondo un paradigma di *infrastruttura a chiave pubblica decentralizzata* (*DPKI*) in cui il Distributed Ledger è la *Root of Trust*.

Il *DID document* contiene le informazioni crittografiche pubbliche e i metodi di verifica per dimostrare il controllo del DID (cioè il controllo delle informazioni crittografiche private) come elemento centrale della creazione del canale sicuro. Il modello di dati del documento DID è codificato utilizzando un formato JSON.

Il livello 2 sfrutta la tecnologia DID (cioè il fondamento della sicurezza del framework SSI) per avviare la procedura di autenticazione.

Infine il **livello 3** (si veda ancora la Fig. 1) *finalizza l'autenticazione, concedendo l'autorizzazione ai servizi* e l'accesso alle risorse utilizzando credenziali dette *verificabili* (*Verifiable Credentials* o *VC*). Le VC costituiscono un meccanismo per esprimere le credenziali digitali in modo crittograficamente sicuro e verificabile da un elaboratore. Una *Verifiable Credential* è un documento digitale non falsificabile che contiene elementi dell'identità digitale aggiuntive rispetto ad una semplice coppia di chiavi e a un DID. Essa contiene informazioni analoghe a quelle offerte da una credenziale fisica, ma garantendo maggiore affidabilità grazie all'aggiunta delle firme digitali che rendono evidenti eventuali manomissioni.

La combinazione di *coppie di chiavi*, una *DID*, un *DID Document* e almeno una *Verifiable Credential* costituisce l'*identità digitale nel framework SSI*. Il livello 3 si basa sul classico triangolo della fiducia (illustrato nella Fig. 1), che nella configurazione classica prevede la coesistenza di tre ruoli diversi:

- Titolare: è l'attore del sistema (utente umano o oggetto) che possiede uno o più VC e che genera appropriate *Presentazioni Verificabili*

(*Verifiable Presentation* o *VP*) ad un *Verificatore* per richiedere accesso ad un servizio o a una risorsa

- Emittente: è l'attore del sistema che emette dichiarazioni su uno o più soggetti, creando una *Verifiable Credential* a partire da tali dichiarazioni e trasmettendola ad un Titolare. Il soggetto è l'attore al quale si riferiscono le dichiarazioni. Molto spesso il Titolare e il soggetto coincidono, ma in alcuni casi il Titolare può anche possedere una credenziale che contiene dichiarazioni relative ad un soggetto diverso;
- Verificatore: è l'attore del sistema che riceve una o più Presentazioni Verificabili (VP) da elaborare (ad esempio eseguendo la verifica crittografica sulle VC generate da Emittenti e Titolari).

Una Credenziale Verificabile contiene metadati per descrivere le proprietà della credenziale, come il contesto, l'ID, il tipo, l'emittente, la data di emissione e di scadenza. Particolarmente importanti tra questi metadati sono le dichiarazioni (una o più, generate dall'Emittente) che il Verificatore utilizza per concedere l'accesso al servizio o risorsa richiesti. Una dichiarazione è una affermazione (di qualsiasi tipo) su un soggetto, espressa da una relazione soggetto-proprietà-valore. L'Emittente esegue una prova crittografica per rendere il VC un documento digitale non falsificabile e verificabile per affermare proprietà / caratteristiche / funzionalità.

Il Titolare richiede al Verificatore l'accesso a servizi e/o risorse mediante una Presentazione Verificabile (VP). La VP è costruita come un involucro di una credenziale verificabile, e contiene la prova crittografica eseguita dal Titolare.

Al momento della generazione di una credenziale verificabile, l'Emittente può anche pubblicare sul ledger uno schema di dati, una sorta di *template* utile per imporre una struttura e un formato di dati specifici per quella VC. In fase di verifica il Verificatore può recuperare tale schema di dati e utilizzarlo per verificare se la struttura e i contenuti di una VC sono ben formati (cioè, sono conformi al template pubblicato).

Gli emittenti sono anche responsabili della revoca delle Verifiable Credential per l'integrità crittografica e di eventuali cambi di stato. Un verificatore, mentre controlla l'autenticità delle VC e delle VP, è tenuto a controllare lo stato della VC e respingere qualsiasi richiesta nel caso in cui la VC risulti revocata. Gli emittenti sono invitati ad annunciare le revoche inserendole in un elenco al quale possono poi accedere i verificatori. La raccomandazione del W3C esorta gli emittenti a utilizzare meccanismi che attenuino le potenziali violazioni della privacy, ad esempio utilizzando un identificatore unico a livello globale come soggetto per una determinata Credenziale Verificabile, e a non riutilizzarlo mai.

A partire dai tre livelli illustrati in precedenza è possibile costruire qualsiasi ecosistema di interazioni affidabili tra esseri umani e cose.

3. La fiducia negli oggetti connessi: *Embedded Trusted Computing*

La tecnica del Trusted Computing è utilizzata da tempo per assicurare che il software che viene eseguito su un elaboratore sia affidabile, ovvero sia certificato dal fornitore e non sia stato sostituito o manomesso. Basata anch'essa sui principi della *crittografia asimmetrica*, la tecnologia Trusted Computing prevede la presenza, a bordo dell'elaboratore che si intende rendere sicuro, di un elemento sicuro denominato TPM (Trusted Platform Module), che svolge numerose funzioni crittografiche utili ad assicurare che non venga eseguito un programma che risulta manomesso o installato senza autorizzazione. Appare evidente come una architettura dotata di modulo TPM rappresenti una soluzione per molte delle criticità dei sistemi che ne sono dotati, impedendo ad esempio la propagazione di un virus o la manomissione di un modulo software presente sul server o sul dispositivo. Il prezzo da pagare è una complicazione architetturale.

Quando la tecnica del Trusted Computing viene applicata ad un oggetto connesso (in molte applicazioni industriali chiamato *Sistema Embedded*), il paradigma risultante viene denominato *Embedded Trusted Computing*

L'obiettivo finale dell'applicazione del Trusted Computing è quello di consentire a un nodo di *misurare e dimostrare la propria integrità* in modo crittografico, cioè di provare agli altri nodi che il software che ha in esecuzione è quello previsto e non è stato manomesso. L'identificazione dei componenti software avviene attraverso una *catena di trust*, che ha come radice una à*ncora di fiducia* (*Trust Anchor* o *TA*). La TA è una entità autorevole per la quale la fiducia è data per scontata e non derivata. Un esempio tipico è la porzione di firmware piccola e accuratamente progettata, che viene eseguita dal sistema all'accensione o al reset dell'hardware. La *Trust Anchor* è responsabile dell›inizio della costruzione della catena di fiducia durante la fase di *bootstrap* del nodo. Ogni elemento della catena è un componente software che, una volta eseguito, svolge i suoi compiti e poi identifica, carica e infine esegue il componente software successivo.

La decisione sulla fiducia può avvenire in qualsiasi fase del processo di *bootstrap* per prendere le giuste contromisure in caso di attacco. Questa decisione può eventualmente essere presa quando il *bootstrap autenticato* è completato; in questo caso si rende necessaria la presenza di un nodo detto *Attestatore Remoto*, responsabile di verificare con una determinata periodicità l'affidabilità dei nodi

target sottoponendo loro opportune *sfide*. La attestazione remota viene eseguita chiedendo al nodo da attestare di inviare, per eseguire le necessarie verifiche crittografiche, informazioni sui componenti software caricati.

In particolare questo sistema è utile nel contesto della Internet of Things, nel quale gli oggetti connessi rappresentano una componente debole a causa delle limitate risorse di calcolo e di comunicazione rispetto a server tradizionali. La presenza di un sistema esterno al quale gli oggetti, una volta terminata la fase di *bootstrap* (eventualmente già resa sicura grazie alla presenza di un TPM), si devono *presentare* dimostrando la propria integrità dal punto di vista sia HW sia SW, offre garanzie di sicurezza superiori a quelle che si otterrebbero con la sola esecuzione di verifiche a bordo dei dispositivi stessi: solo dopo aver superato con successo la fase di *Remote Attestation* i dispositivi vengono autorizzati ad accedere alle risorse che fanno parte del contesto nel quale sono destinati ad operare.

4. *Cybersecurity by Redundancy*: architetture distribuite e ridondanti

L'ultimo esempio di soluzione architetturale che presentiamo come strumento per aumentare la sicurezza informatica è basata su un principio consolidato e semplice da comprendere: la *ridondanza*, ovvero la replica delle informazioni su diversi sistemi informatici. Introducendo ridondanza si pone chi volesse attaccare un sistema o un servizio per distruggere o modificare dati e informazioni nelle condizioni di dover eseguire l'attacco su una molteplicità di sistemi informatici, affrontando quindi una complessità molto maggiore. È immediato comprendere come la sicurezza che si ottiene con questa tecnica è tanto maggiore quanto più elevato è il numero di elaboratori sui quali le informazioni sono replicate.

Dal punto di vista tecnico la difficoltà nella gestione di un sistema di questo tipo, nel quale non esiste un sistema eletto a svolgere le funzioni di orchestratore o arbitro (che sarebbe un punto di debolezza dell'intero sistema) ma solo elaboratori indipendenti tra loro, consiste nell'assicurare che tutti mantengano aggiornata la propria replica dei dati da rendere sicuri e affidabili. Questa esigenza ha fatto nascere e crescere un'area di studio e ricerca tecnologica di grande complessità denominata *protocolli di consenso nei sistemi distribuiti*.

Il consenso tra un gruppo di elaboratori che collaborano svolgendo le stesse funzioni viene raggiunto quando la maggioranza assoluta (50% + 1) dei partecipanti concorda su una determinata decisione (che può anche semplicemente essere un valore numerico, o la versione di un documento da considerare

valida e ufficiale). I sistemi distribuiti e le tecniche di consenso sono argomenti studiati nella *Computer Science* fin dagli anni '70 dello scorso secolo, e hanno avuto recentemente una nuova fase di evoluzione importante con la introduzione dei sistemi denominati *Distributed Ledger*: in alcune di queste complesse architetture (la più famosa delle quali è la *BlockChain*) i sistemi partecipanti (diverse migliaia o decine di migliaia di elaboratori) concorrono nel mantenere aggiornato un registro delle transazioni al quale vengono continuamente aggiunti i nuovi movimenti. Una componente importante della sicurezza offerta dai sistemi di Distributed Ledger risiede nel fatto che un attaccante che volesse modificare le informazioni contenute nel registro delle transazioni dovrebbe violare migliaia di elaboratori, apportando la modifica su ciascuno di essi.

Conclusioni

La combinazione delle tre tecniche presentate, ovvero Identità Digitale distribuita, Trusted Computing e DLT permette di sviluppare sistemi connessi distribuiti nei quali ogni elemento può verificare la fiducia che ripone in altri componenti e nei dati scambiati prima di prendere decisioni operative critiche. Questo approccio è fondamentale – e oggetto di specifici *trend* di ricerca – nel caso delle infrastrutture critiche, per le quali il concetto di *fiducia* nei dispositivi elettronici installati è fondamentale per operare in modo sicuro garantendo l'incolumità delle persone che convivono con le infrastrutture digitali.

Le tre tecniche presentano caratteristiche funzionali e costi differenti, che le rendono più o meno adatte a seconda dei possibili scenari nei quali possono essere impiegate.

La soluzione che presenta la più ampia applicabilità è la gestione della *identità digitale distribuita*, che rappresenta un concetto applicativo e trasversale, e che può essere implementato utilizzando soluzioni tecnologiche diverse. La assegnazione di una identità a ciascuno degli attori che partecipano ad una applicazione digitale (siano essi umani, elaboratori o dispositivi embedded) è una *conditio sine qua non* in quanto la conoscenza degli attori di una qualsiasi interazione è il punto di partenza per poter implementare meccanismi di sicurezza in ogni sistema distribuito. Tale conoscenza consente infatti di eseguire verifiche di autorizzazione e soprattutto di attribuzione di responsabilità per ciascuno degli eventi di interesse rilevati.

La implementazione di soluzioni di *Trusted Computing* richiede una revisione della architettura elettronica dei sistemi target, e pertanto risulta adatto

a sistemi che presentano una significativa criticità in termini di verifiche di cybersecurity: componenti che intervengono nel controllo di infrastrutture critiche (sensori / attuatori) o schede installate a bordo di robot, veicoli o velivoli rappresentano certamente esempi di dispositivi per i quali gli elevati costi di riprogettazione HW sono commisurati ai vantaggi ottenuti. Più in generale queste considerazioni valgono in tutti i casi nei quali attacchi di tipo cyber potrebbero avere effetti nel mondo fisico mettendo a rischio la incolumità (*safety*) delle persone (automotive, velivoli, veicoli autonomi, robotica di servizio, etc).

Infine, l'uso delle *DLT* interviene a complemento delle altre soluzioni descritte, in quanto tali tecnologie forniscono servizi di memorizzazione immutabile dei dati, caratteristica strumentale alla implementazione di soluzioni di identità digitale e verifica della integrità del software e dei dati. Essendo inoltre infrastrutture pubblicamente accessibili e disponibili, il loro impiego può risultare conveniente anche dal punto di vista degli investimenti necessari alla realizzazione delle soluzioni.

Riferimenti bibliografici

D. Kushner, *The Real Story of Stuxnet*, IEEE Spectrum, Vol. 50, Issue 3, 2013.

Trade-offs between Distributed Ledger Technology Characteristics, ACM (Association for Computing Machinery) Computing Surveys, Vol. 53, No 2, 29 marzo 2020.

F. Chiarello *et al.*, *Analisi del caso BlackEnergy: tra geopolitica e cybersecurity*, Analytica for Intelligence and Security Studies, giugno 2021.

TCG TSS 2.0 Overview and Common Structures Specification, TCG (Trusted Computing Group), Version 1.0, 23 settembre 2021.

S.M. Kerner, *Colonial Pipeline hack explained: Everything you need to know*, Techtarget.com, 26 aprile 2022.

Decentralized Identifiers (DIDs) v1.0: Core Architecture, data model and representations, W3C Recommendation, 19 luglio 2022.

Verifiable Credentials Data Model v1.1, W3C Recommendation, 3 marzo 2022.

Distributed Ledger Technologies: Research and Practice, ACM (Association for Computing Machinery), Vol. 2, Issue 1, marzo 2023.