Il Futuro della Sicurezza IT tra “Consumerization” e “Embedding”

Sommario

Periodicamente nuove parole diventano di moda, una delle ultime è “Consumerization” che insieme a “Cloud”, “BYOD” ecc. danno comunque una indicazione degli ultimi sviluppi dell'IT. E' di particolare interesse analizzare quanto possa influire sulla sicurezza IT il trend attuale dello sviluppo del mercato IT sempre più dedicato alla produzione di device “Commodity” per l'utente finale ove il “Embedding” del software è necessario, e l'interazione dell'utente con il device è sempre più ad alto livello.

E' utile periodicamente riflettere sui mutamenti in atto nel mondo dell'IT e della sicurezza IT per imparare dalle esperienze passate e cercare di dare una occhiata a quello che ci potrebbe aspettare nel prossimo futuro. Oltre al “Cloud” di cui non parleremo, di “BYOD”, di “Device Embedded” ecc., l'ultima moda in IT è la “Consumerization”, parola in giro da una decina di anni, ma che solo recentemente ha incominciato ad essere utilizzata anche dal Marketing, il che vuol dire che ha raggiunto ormai la maturità.

Tra Commodity e Consumerization

Consumerization è l'altra faccia della Commodity: se i prodotti e servizi IT Commodity sono semplici, di basso costo ed uguali per tutti, la Consumerization è l'orientamento degli stessi prodotti e servizi verso l'utente individuale. Le due cose non sono in contraddizione, anzi, ma vivono in un, a volte precario, equilibrio. Partendo da una base di strumenti Commodity, che garantisce bassi costi, ampie funzionalità e grande distribuzione, si aggiunge una personalizzazione che fidelizza l'utente e gli permette di utilizzare gli strumenti IT nel modo a lui più utile e piacevole.

In pratica il concetto è abbastanza semplice: abbiamo tutti gli stessi strumenti hardware, sistemi operativi e applicazioni, ma è lasciata la libertà ad ogni utente di scegliere le “App” che preferisce, di apportare configurazioni (dalle suonerie agli sfondi ecc.) personalizzate, in modo che il proprio device IT diventi unico ed individuale.

Ma ovviamente una volta che ognuno di noi ha un device IT unico e personale, diventa scomodo e difficile usarne un altro, ad esempio per lavoro. Ecco quindi che dalla Consumerization si arriva al

BYOD, ovvero “Bring Your Own Device” al lavoro.

Ma prima di analizzare le conseguenze di tutto ciò, conviene riflettere su quanto successo negli ultimi anni, grazie soprattutto alla “visione” di Apple e del suo iconico fondatore, Steve Jobs. Tutti eravamo convinti che l'informatica per diventare di massa dovesse semplificarsi, diventare sempre più un “oggetto” di uso quotidiano, quanto una televisione, un frigorifero od una automobile. Solo che quasi tutti partivamo dal punto di vista sbagliato, cercare di modificare e far evolvere i computer sia dal punto di vista hardware che software.

Invece l'approccio vincente è stato quello di partire dal telefono (cellulare) sino ad allora strumento con ben poche risorse e possibilità (ad eccezione della messaggistica avanzata introdotta ad esempio da Blackberry per l'utenza aziendale più sofisticata). Partire dal cellulare ha dato mano libera nel ridisegnare l'interfaccia e nello stabilire dei nuovi paradigmi di interazione tra utente e strumento.

E' nato così il mondo delle “App”, ove l'interazione tra utente e strumento si sposta ad un livello molto più alto nella pila software rispetto al computer tradizionale. L'utente non ha accesso ne al sistema operativo ne al middleware applicativo. Ha solo una interfaccia personale che gli permette di scegliere, scaricare, installare e personalizzare le applicazioni di suo interesse. Tutto il resto è nascosto all'interno del prodotto: è si software, ma per l'utente è come se fosse hardware, ovvero non ci può accedere direttamente (a meno di jailbreaking o simili violazioni dei sistemi).

Il mondo delle “App” è sicuramente stato il grimaldello che ha permesso di passare da Commodity a Consumerization, liberando l'utente da tutti quei compiti tecnici e tecnologici in realtà non di sua competenza.

Così oggi abbiamo smartphone, tablet ed anche, almeno in linea di tendenza, computer (portatili o da tavolo) che ci permettono di gestire le “App”, creare un nostro strumento personalizzato con i nostri soli dati, e dimenticarci delle complicazioni sottostanti.

Diventa quindi molto interessante, utile sia per la persona che per le aziende, utilizzare questi stessi strumenti sia per scopi personali che di lavoro.

I dati e la sicurezza

Ovviamente e come la storia dell'IT ci insegna, le problematiche di sicurezza si comprendono, studiano ed affrontano solo dopo che i sistemi sono in mano agli utenti finali. Questo rende ovviamente difficile evitare piccoli, grandi ed enormi problemi di sicurezza.

Il primo problema da affrontare è la gestione della sicurezza dei diversi tipi di dati presenti su questi strumenti IT. In primo luogo bisognerebbe distinguere tra informazioni personali e aziendali, che nella maggior parte dei casi sono di proprietà di entità diverse e soggette a regole, vincoli e requisiti di sicurezza diversi. La discussione su questo punto è aperta, si veda [1] ad esempio, e almeno nelle

aziende medio/grandi con un minimo di sensibilità per le problematiche di sicurezza IT, il problema è noto, affrontato od almeno discusso.
Vista la In-Sicurezza in generale di questi strumenti, molto spesso l'approccio scelto è ancora quello di dotare il lavoratore di uno strumento aziendale, quindi sottoposto e gestito secondo le politiche aziendali e di non permettere l'uso del BYOD, se non in casi particolari o eccezionali.

Ma il problema ancora più stringente è che la Consumerization degli strumenti IT, rendendo ogni strumento personale, individuale ed unico, pone l'onere della protezione dei dati trattati dallo strumento nel suo utilizzatore finale. Infatti anche ipotizzando che le App e tutto il software ed hardware forniscano ogni possibile funzionalità di sicurezza, è solo l'utilizzatore finale che è in grado di valutare il livello di sicurezza e quali protezioni debbano essere applicate ai dati secondo il loro contesto. Ad esempio, le informazioni di geolocalizzazione dello strumento stesso (ormai quasi tutti gli strumenti IT, tramite GPS, triangolazioni GMS/GPRS/WiFi, geolocalizzazione IP ecc., sono in grado di stabilire più o meno precisamente la propria posizione sul globo terreste) possono essere dati utili o addirittura indispensabili per certe applicazioni (ad esempio lavorative) e al contrario addirittura pericolose quando condivise con altre applicazioni (ad esempio social networking ecc.). Solo l'utente è in grado potenzialmente di valutare il livello di protezione da applicare ad ogni informazione dipendente dal contesto di utilizzo sia della persona (al lavoro, in famiglia, con gli amici ecc.) che dell'applicazione.

Questo è ovviamente impossibile anche per chi è conscio dei rischi e delle problematiche di sicurezza. Ancora oggi, purtroppo, l'utente medio di uno strumento IT non è assolutamente in grado non solo di gestire la sicurezza delle proprie informazioni ma di apprezzarne il valore e comprendere la differenza tra una battuta al Bar ed uno scherzo postato su di un social network pubblico. La sensibilità alla “Privacy” dei dati scambiati o resi pubblici sugli strumenti informatici è ancora molto lontana per l'utilizzatore medio che è esposto solo da pochi mesi, al più qualche anno, al significato ed alle conseguenze di rendere pubblici ed incancellabili i propri dati.

Sicurezza IT e Privacy

Dobbiamo però chiarirci ulteriormente le idee, in quanto il confine tra Sicurezza e Privacy è spesso labile e poco chiaro, soprattutto nel campo dell'IT. Tipicamente la Privacy concerne la limitazione o il controllo della divulgazione di informazioni di carattere personale, raramente di carattere aziendale. Se una persona rende pubbliche proprie informazioni volutamente, non vi può essere alcun problema di sicurezza, al più di conoscenza da parte dell'individuo delle conseguenze dell'atto

che ha compiuto. Chi fornisce ad una persona uno strumento che potrebbe arrecare danni diretti o indiretti a chi lo utilizza anche tramite la divulgazione di informazioni personali, dovrebbe almeno in linea di principio informare l'utilizzatore dei rischi a cui potenzialmente va in contro, oltre che ovviamente affermare la propria “esclusione di responsabilità” (Disclaimer) per qualunque tipo di uso non “accorto” dello strumento.
Benché in questo ambito i rischi siano molto grandi, ben poco si può fare dal punto di vista tecnico della sicurezza IT: il “Fattore Umano” in questo caso è l'elemento preponderante.

Dal punto di vista tecnico della sicurezza IT, l'unica cosa che si può fare per rendere più “sicuro” l'utilizzo di questi strumenti al tempo della Consumerization è quello di fornire a tutti i livelli, hardware, sistemi operativi, piattaforme applicative (middleware) e applicazioni, funzionalità che permettano di gestire nella maniera più facile ed intuitiva possibile, le caratteristiche di sicurezza dei dati. Sappiamo bene quanto difficile sia realizzare funzionalità di sicurezza che siano al contempo facili da usare, intuitive e efficaci. Per l'utente finale, anche esperto, la maggioranza delle funzionalità di sicurezza sono contro-intuitive, difficili da utilizzare e troppo limitanti dell'uso dello strumento: alla fine appaiono solo come un impedimento con poco scopo se non nullo.

Embedding e bugs

Ma l'altro aspetto tecnico della sicurezza IT è quello intrinseco dell'IT stesso. Abbiamo ben detto che un notevole cambiamento recente è stato quello della separazione tra il livello non-tecnico dell'utente che sceglie, configura ed utilizza le proprie App, e tutta la tecnica IT, hardware e software, che non è gestibile direttamente dall'utente. Questo non è un approccio diverso da quanto succede ogni giorno per frigoriferi, televisioni, automobili ecc.: l'utente utilizza questi strumenti ma non ha accesso al loro funzionamento interno che è lasciato, nei rari casi rimasti, solo al personale specializzato.
Possiamo chiamare questo come un processo di “Embedding” dell'IT, ove sempre un maggior numero di componenti di un sistema IT viene a far parte della componente “embedded” nell'hardware dello strumento. Questa è sicuramente la direzione giusta e l'evoluzione che ci aspettiamo da parte dell'IT se non che ha un grave problema: chi gestisce e come sono gestiti i “bug” (vulnerabilità) del software? Questo è un argomento dibattuto (si veda anche solo [2] e [3]) ma conviene analizzarlo un poco in dettaglio.
Consideriamo come primo esempio quello delle piattaforme degli smartphone. Come abbiamo detto, lo smartphone nasce come evoluzione del telefono e non come evoluzione del personal computer. Come tale la sua impostazione è quella di tutti i device da usare, click-and-go: non è

prevista la manutenzione del software di base, solo la gestione delle App. Questo però assume che il software di base non abbia problemi di sicurezza e vulnerabilità che richiedano il suo aggiornamento. Ovviamente nell'approccio consumistico a commodity, la mancanza o carenza di funzionalità fa parte del gioco del mercato: invece di fare un upgrade software del sistema, si compra il nuovo modello di smartphone che fornisce le funzionalità volute. Questo anche quando l'hardware del vecchio modello sarebbe stato in grado tramite un upgrade software, di fornire le nuove funzionalità. Non è però conveniente a livello di mercato fornire questa possibilità, conviene far comprare agli utenti un nuovo modello piuttosto che gestire quello vecchio. Questo non solo rende il mercato più vivo, ma semplifica anche la catena della gestione dei prodotti. Infatti i produttori sono impegnati solo a produrre i nuovi modelli e non a mantenere quelli vecchi, riducendo di molto la struttura organizzativa e soprattutto i costi finali per se stessi e per i consumatori.

Il modello consumistico usa-e-getta è sicuramente ottimo per velocizzare lo sviluppo dei prodotti e ampliare al massimo in breve tempo il numero di utenti. Ha l'ovvio problema della gestione e smaltimento dei device usati (o guasti) dovuta alla produzione di grandi quantità di device "vecchi" in brevi tempi. Ma dal nostro punto di vista ha un problema essenziale: la mancanza di supporto nel caso di vulnerabilità o problemi di sicurezza del software di base. Mancando una vera architettura di supporto e gestione del software, è praticamente impossibile per i produttori gestire gli update di sicurezza del software. Il che ovviamente lascia gli utenti indifesi di fronte ai problemi di sicurezza che ormai sono quotidiani.

Il rapidissimo sviluppo dell'hardware rende la situazione ancora più complessa. Ogni modello di smartphone (si pensi in particolare come esempio all'ecosistema di Android) ha il proprio hardware che richiede driver dedicati esclusivamente per quell'hardware. Il modello successivo ha qualche componente hardware diversa per le quali i driver sono modificati opportunamente. Un patch di sicurezza del software di base deve essere compatibile con tutte le migliaia di modelli diversi e diverse personalizzazioni di driver ecc., ed è in pratica oggi impossibile o sicuramente troppo oneroso, gestire e garantire questa compatibilità. In pratica oggi il migliore approccio per un utente in presenza di una vulnerabilità di sicurezza del proprio smartphone è quello di cambiare smartphone e acquistarne uno con una versione del software immune dai problemi di sicurezza noti.

Si confronti questa situazione con quella a cui siamo abituati nei personal computer. In questo caso l'approccio è esattamente il contrario: l'utente deve impegnarsi in prima persona a gestire sia la sicurezza che gli aggiornamenti di tutto il software. Deve installare e utilizzare prodotti di sicurezza quali anti-virus ecc., e deve aggiornare periodicamente tutto il software presente sul proprio personal computer. Come abbiamo visto, nelle piattaforme smartphone l'utente è sgravato da queste

responsabilità che in realtà non gli toccherebbero, ma al contempo nessuno sinora si è preso in carico la responsabilità di gestire per lui gli aggiornamenti del software soprattutto quelli relativi alla sicurezza.

Certo si parla di creare anche su queste piattaforme strumenti di sicurezza quali anti-virus ecc. o processi di aggiornamento del software (di base) periodico (gli aggiornamenti delle App sono quasi sempre automatici e gestiti), ma come detto l'impostazione e la configurazione di queste piattaforme rende molto difficile tutto ciò. Ad esempio, non ha senso che un anti-virus sia una App, in quanto per poter garantire un reale servizio di sicurezza deve essere una componente del software di base. Infatti per definizione una App non ha accesso a dati e informazioni tali da poter gestire la sicurezza delle altre App e quindi nessun reale strumento di sicurezza può essere fornito come App stessa.

Sempre più Embedding

Ma un secondo ambito ove la sicurezza IT diventa quotidianamente di maggior importanza, è in molti altri device IT connessi a Internet. A parte citare i frigoriferi, che periodicamente tornano di moda ma sinora senza una vera ragione, quelli che oggi preoccupano sono gli innumerevoli device connessi a Internet e non gestiti dall'utente finale, ovvero delle scatole che funzionano da sole. Portiamo solo due esempi: i router e i televisori.

Recentemente vi sono stati molti annunci di problemi di sicurezza di router sia casalinghi (Adsl ecc.) che anche router in utilizzo da provider (ISP). Ormai questi device sono nelle case di tutti, nelle centraline telefoniche agli angoli delle strade ecc. ed è facile capire che i loro numeri sono veramente grandi. La maggior parte di questi device è costruita partendo da componenti hardware prodotti in serie ai quali sono adattate per lo più vecchie versioni di Linux con driver personalizzati per le componenti hardware particolari. La possibilità di gestione degli aggiornamenti del software di base, con i problemi di compatibilità con le versioni specifiche dell'hardware come nel caso degli smartphone, è anche in questo caso estremamente difficile se non praticamente nulla. Visto che in molti casi non è neanche prevista la possibilità di aggiornare il software, cosa possiamo fare quando viene scoperta una vulnerabilità grave di sicurezza?

Oggi non sappiamo come rispondere a questa domanda, e corriamo sicuramente dei gravi rischi. Recentemente vi sono stati casi in cui vulnerabilità di router casalinghi o di ISP sono state sfruttate per portare ad esempio attacchi di Denial-Of-Service (ad esempio via DNS o NTP [con il comando 'monlist']), oppure per accedere a reti interne od a traffico riservato.

Ma non sono solo i router, i personal computer e gli smartphone che sono connessi a Internet. Negli ultimi anni anche i televisori per offrire maggiori possibilità agli utenti, hanno integrato l'accesso a dati, ed in alcune aree anche a trasmissioni video, via internet. Di nuovo, all'interno del televisore vi

è un piccolo sistema embedded che fornisce sia il software di base che le applicazioni (un browser ed una piattaforma per la ricezione di video in streaming) e la connettività (tipicamente WiFi e Ethernet). Anche in questo caso, gli aggiornamenti del software di questi sistemi non sono di solito previsti e sicuramente sono l'ultima preoccupazione dell'utente. Quando mai abbiamo fatto un update del software del nostro televisore? Ma cosa potrebbe succedere nel caso vi sia (cosa molto probabile) una vulnerabilità di sicurezza nel software utilizzato? Anche in questo caso nella maggior parte dei prodotti sono stati adottati sistemi di software di base semplici, di vecchia data ma molto stabili, noti per essere facilmente adattabili a vari modelli di hardware ma a rischio di avere molte vulnerabilità di sicurezza.

Non sappiamo ancora come affrontare e gestire queste problematiche di sicurezza IT in maniera sistematica. Dobbiamo ancora capire come valutare questi rischi, le possibili conseguenze ed i costi per gestirli nel migliore dei modi. Dobbiamo essere molto attenti da una parte a non minimizzare queste tematiche, ma al contempo anche a non esagerarle in quanto il rapporto costi-benefici è stato, è e sarà fondamentale per lo sviluppo dell'IT e sappiamo bene che la sicurezza IT è sempre costosa, e può diventare talmente costosa da essere anche controproducente per l'IT stesso.

Riferimenti

[1] ENISA “Consumerization of IT: Final report on Risk Mitigation Strategies and Good Practices” http://www.enisa.europa.eu/activities/risk-management/evolving-threat-environment/COIT_Mitigation_Strategies_Final_Report

[2] Andrea Pasquinucci “On the Practical Impossibility of Embedded ICT Security”, http://www.ucci.it/docs/CFS-200811.pdf

[3] Bruce Schneier “Security Risks of Embedded Systems”, https://www.schneier.com/blog/archives/2014/01/security_risks_9.html

Andrea Pasquinucci (PhD CISA CISSP)