Economia della Cyber security

Vi sarà capitato di sentire che “gli antivirus funzionavano fintanto che erano gli stessi produttori di antivirus a creare i virus informatici”.

Anche se si tratta di una chiacchera da bar, questa frase rispecchia il cambiamento di un thread model globale; da un lato ci sono persone che si occupano per lavoro di difendere dalle minacce e dall’altro ci sono persone che per lavoro le creano.

Se tra queste ultime escludiamo quelle che agiscono per fini politici, possiamo limitarci a considerare solo gli attacchi in cui il fine ultimo non è danneggiare ma avere un ritorno economico.

Possiamo analizzare come funziona questa “cyber” economia e alla fine domandarci:

“Esiste un paradigma di cyber-economia per cui l’attacco sia possibile ma non economicamente vantaggioso?”

Oppure potremmo domandarci: “Se riuscissimo ad incentivare chi produce nuovi virus a fornire la soluzione all’infezione prodotta, potremmo diminuire le vittime di attacchi zero-day?”

L’economia della sicurezza informatica oggi

Oggi si usa parlare di “cyber-crime as a Service”. Questo nuova economia spinge un numero limitato di persone molto capaci a creare nuovi exploit che poi venderanno sul dark web facendo profitti.

Dall’altra parte, un numero limitato di aziende nel settore della sicurezza informatica saranno spinte a creare nuove protezioni per quegli exploit che poi venderanno facendo

profitti. Negli ultimi anni entrambe le parti hanno aumentato i profitti a scapito degli utenti finali, che si trovano a pagare sia il costo della protezione sia quello del danno.

Un problema di fiducia: il caso bitcoin

L'attuale paradigma di sicurezza informatica si basa su un modello centralizzato.

Mi spiego meglio: l'utente si affida ad un'azienda di sicurezza informatica, paga un abbonamento mensile, e riceve in cambio aggiornamenti software che servono a proteggerlo dalle minacce. Ci tengo a sottolineare che l'utente paga l'azienda non l'aggiornamento.

In altre parole l'aggiornamento è valido solo se ricevuto da una fonte fidata. D'altra parte secondo l'attuale paradigma non ci sarebbe modo di distinguerne uno valido da uno malevolo.

Ora proviamo ad immaginare un paradigma diverso dove la validità dell'aggiornamento non è data dalla fonte da cui proviene. Come si può fare diversamente?

Il problema che si presenta è analogo al problema affrontato da "Satoshi Nakamoto" quando creò il bitcoin; serve un protocollo del consenso.

Se trascendiamo dall'implementazione, un protocollo del consenso è un meccanismo che sostituisce un principio di fiducia con un principio di incentivo/disincentivo.

Un'industria decentralizzata per la sicurezza informatica

L'avvento della blockchain apre la strada a nuovi modelli di business. Lo scopo di questo articolo è quello di promuovere un ragionamento che stimoli il lettore a pensare nuovi modelli di business per l'industria della cyber-security.

Riprendiamo il ragionamento fatto. Abbiamo approcciato lo scenario attaccante – difensore sotto il profilo economico. Abbiamo ipotizzato che i due attori agiscano per aver un ritorno economico. Abbiamo sottolineato come pochi grandi colossi dell'industria della sicurezza informatica si trovino oggi in difficoltà nel fronteggiare minacce sempre più numerose ed imprevedibili.

Ora stiamo valutando l'ipotesi di un sistema decentralizzato dove chiunque (e non solo i pochi grandi colossi) possa guadagnare promuovendo i meccanismi di protezione da virus informatici. Perché?

Perché se colui che scopre un nuovo tipo di attacco guadagna di più a proteggere gli utenti che a rivendere l'attacco (cyber-crime as a service), forse il numero di vittime diminuirebbe.

Tuttavia un sistema decentralizzato, anche detto DAO – decentralized autonomous system, deve essere costruito su un insieme di regole (scritte nel codice) che mantengano il sistema funzionale senza l'intervento di un'autorità centrale.

In particolare dobbiamo trovare un sistema che incentivi chi inserisce l'aggiornamento lecito e che disincentivi chi inserisce l'aggiornamento truffaldino.

Un aggiornamento per essere valido deve:

1. riferirsi ad una minaccia esistente: occorre stabilire se una minaccia è realmente una minaccia. In questo caso diremo che la minaccia è vera.
2. avere una soluzione che funziona: l'aggiornamento proposto deve bloccare quella minaccia. In questo caso diremo che la soluzione è vera.

occorre quindi:

1. incentivare chi scopre una vera nuova minaccia /

soluzione e disincentivare chi scopre una falsa nuova minaccia / soluzione
2. incentivare chi valida una vera nuova minaccia / soluzione e disincentivare chi valida una falsa nuova minaccia / soluzione

Ipotizziamo che io riesca a scoprire una nuova minaccia. Poiché sono sicuro di aver scoperto una minaccia vera, non ho nulla da temere nel far verificare a chiunque ne abbia le competenze la veridicità della mia scoperta.

Per dimostrare la mia buona intenzione decido di buttare via ottocento euro (1 ether). In Ethereum questa azione corrisponde ad inviare l'importo pagato msg.value all'indirizzo nullo.

```
function submitNewThreat(bytes32 proofSoureCodeHash) public payable {             address _beneficiary = 0x0;
_beneficiary.send(msg.value);
```

Questi ottomila euro si trasformano in un numero che indica il livello di fiducia che questa minaccia si è prefissata di raggiungere. Questo numero si calcola dividendo il denaro buttato via per un costo fisso che in questo esempio assumiamo essere di 8 euro (0.001 ether).

```
Mapping (bytes32 => uint) goalTrustLevelForNewThreat;
goalTrustLevelForNewThread = msg.value  / 0.001 ether;
```

In questo caso il livello di fiducia minimo da raggiungere sarà di 100; ovvero il proponente si aspetta che almeno 100 persone valideranno la minaccia trovata entro un arco di tempo limitato. Ovvero:

```
function isValidThreat (proofSourceCodeHash) public payable { require (msg.value == 0.001) currentTrustLevelForNewThread += 1; address _beneficiary = 0x0;             _beneficiary.send (msg.value);
```

Si noti che anche colui che valida l'idea butta via i soldi inviandoli all'address null. Perché dovrebbe farlo? Perché

dopo aver avuto accesso al codice sorgente della minaccia ed averne verificato la veridicità anche lui è disposto a buttare via soldi per dimostrarlo.

Inoltre avendo avuto accesso al codice della minaccia può iniziare a lavorare ad una soluzione e se la trova potrà essere il primo a proporla.

Allo stesso modo con cui è inserita la minaccia, il primo che trova una soluzione e che ritiene sia veritiera inserisce la soluzione buttando via, in questo esempio, 8 mila euro (10 ethet).

In questo caso per validare una soluzione è necessario buttare via 80 euro (0.1 ether) e per ciò saranno necessarie 100 persone paganti.

A questo punto vi starete domandando perché così tante persone hanno bruciato i loro ethers.

Dobbiamo ricordare che nel nuovo sistema ci sono anche gli utenti finali che pagano per l'aggiornamento a patto che l'aggiornamento sia adeguato, ovvero la minaccia e la soluzione siano vere.

Allora fissiamo un soglia numerica che rappresenta il minimo livello di trust che una minaccia deve avere per essere ritenuta attendibile.

Se un aggiornamento supera questo livello allora gli utenti lo installeranno pagando per quello specifico aggiornamento un costo fisso supponiamo di 0.8 euro (0.001 ether).

function getAntiVirusUpdate( )

Questo vuol dire che se poco più di 20k utenti installano l'aggiornamento, tutti quelli che hanno partecipato, circa 200 persone, vanno in pari.

Nel caso gli utenti fossero molti di più quell'aggiornamento

inizierebbe ad andare in attivo ed i guadagni verrebbero divisi secondo quanto ogni partecipante ha versato. Così colui che ha trovato per primo la soluzione avrà il maggior ricavo (quota 1000) e poi a seguire colui che ha trovato la minaccia (quota 100), poi tutti quelli che hanno validato la soluzione (quota 10 ciascuno) e quelli che hanno validato la minaccia (quota 1).

Conclusione

Cosa deve fare un malintenzionato per forgiare un aggiornamento truffaldino?

Nell'esempio sopra deve fare 200 account e buttare via circa 18 mila euro (circa il doppio proposto dai due primi inserzionisti).

Questo attacco è chiamato Sybil attack ed è un attacco noto nei sistemi di reputazione in reti P2P. Naturalmente un sistema di difesa che sia esso stesso vulnerabile non è tollerabile.

Tuttavia lo scopo dell'articolo è quello di stimolare il pensiero creativo del lettore e non dimostrare la fattibilità del sistema in oggetto.

A cura di: **Francesco Ermini**

L'articolo Economia della Cyber security proviene da ICT Security Magazine.

https://www.ictsecuritymagazine.com/articoli/economia-della-cyber-security/