



delphiedintorni.it



 *wintech italia*

SQL INJECTION



 *wintech italia*



Luca MINUTI

DEVELOPER

email

Luca.minuti@gmail.com

GITHUB

[HTTPS://GITHUB.COM/LMINUTI/](https://github.com/Lminuti/)

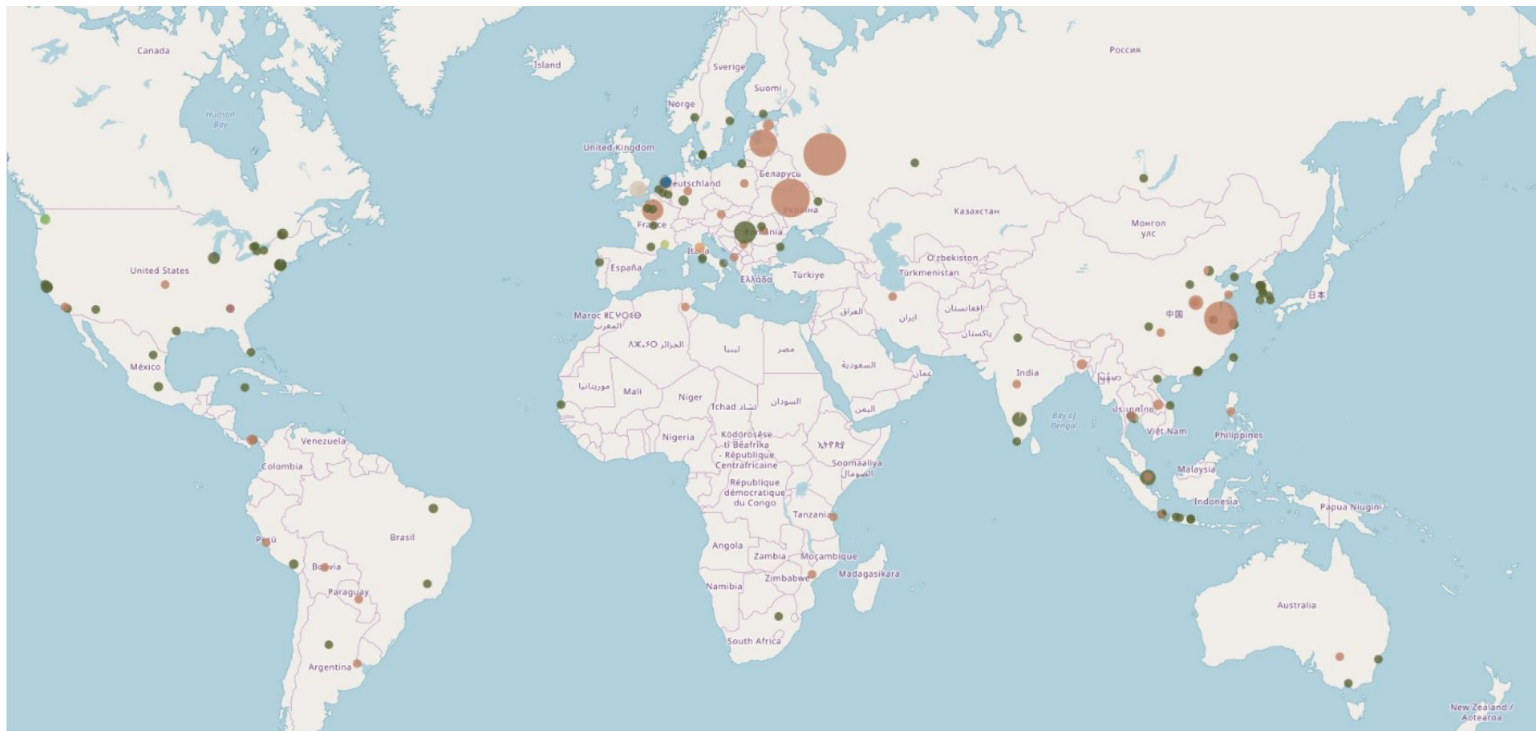


UN PO' DI DATI

- In media, ogni settimana viene scoperta una nuova vulnerabilità zero-day
- Rubate più di mezzo milioni di dati personali
- Importanti vulnerabilità di sicurezza presenti in tre quarti dei siti Web più noti
- Le campagne di spear-phishing che colpiscono i dipendenti delle aziende sono aumentate del 55%
- Il ransomware è aumentato del 35%
- Bloccate cento milioni di truffe del supporto tecnico

https://www.symantec.com/it/it/security_response/publications/threatreport.jsp

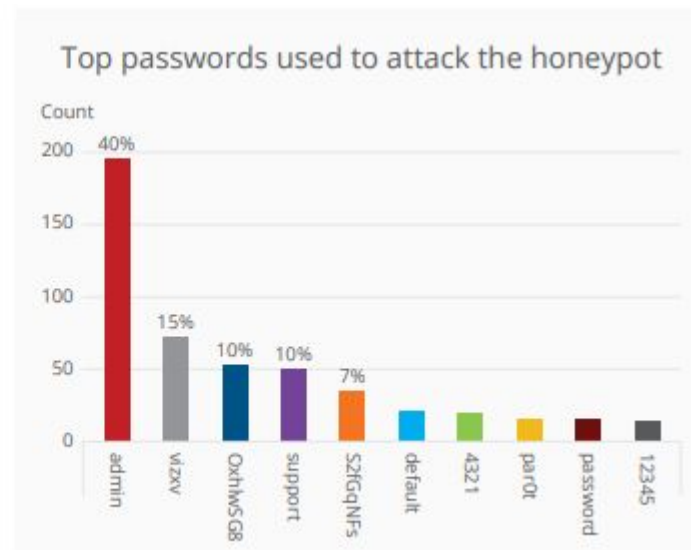
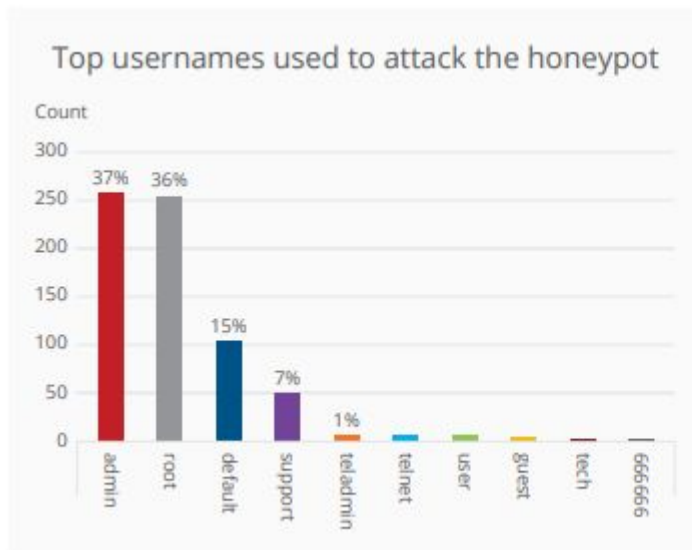
UN PO' DI DATI



McAfee Labs Threat Report August 2019. Figure 6. Global locations that host (likely compromised) malicious activity on all protocols.

<https://www.mcafee.com/enterprise/en-us/threat-center/mcafee-labs/reports.html>

UN PO' DI DATI

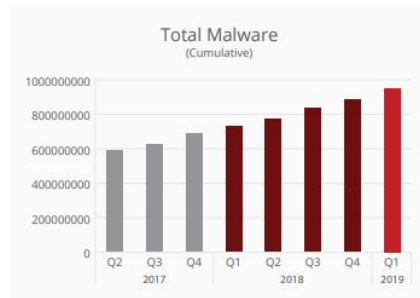


“**Honeypot:** (letteralmente: "barattolo del miele") è un sistema o componente hardware o software usato come "trappola" o "esca" a fini di protezione contro gli attacchi informatici.

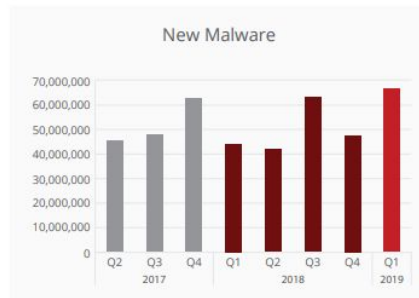
<https://www.mcafee.com/enterprise/en-us/threat-center/mcafee-labs/reports.html>

UN PO' DI DATI

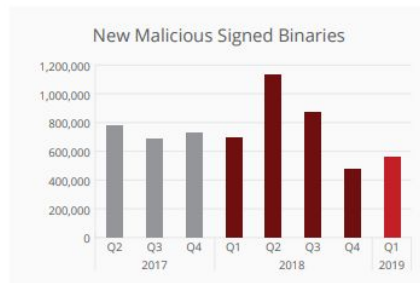
Malware



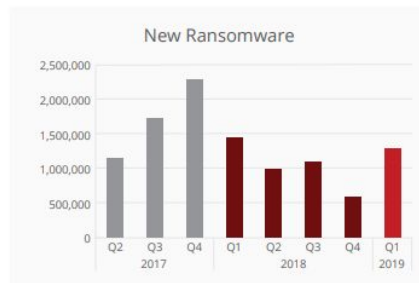
Source: McAfee Labs, 2019.



Source: McAfee Labs, 2019.

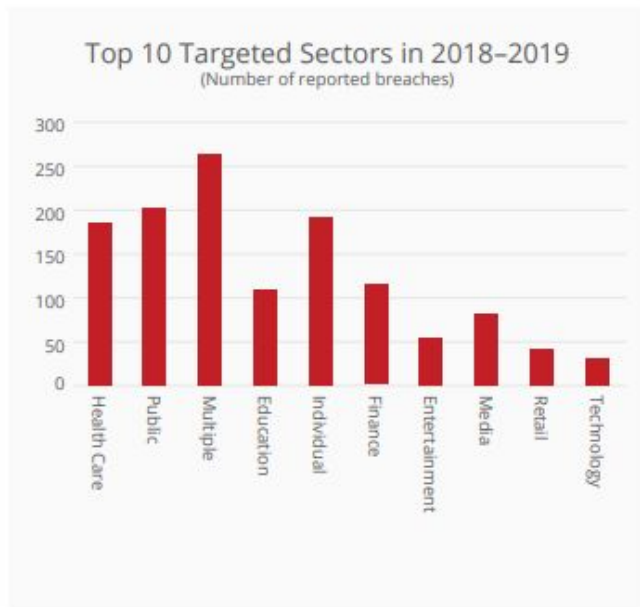


Source: McAfee Labs, 2019.

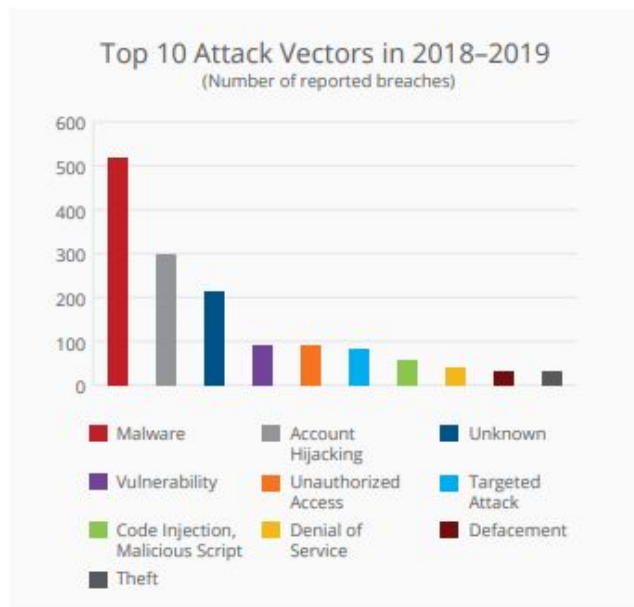


Source: McAfee Labs, 2019.

UN PO' DI DATI



Security incidents data is compiled by McAfee Labs from several sources.



Security incidents data is compiled by McAfee Labs from several sources.

OWASP

- L'Open Web Application Security Project
- Progetto open-source per la sicurezza delle applicazioni
- Nato il 9 settembre 2001
- Guide e consigli sulla creazione di applicazioni Internet sicure
- Suite di test (proxy tool: zap)
- Top 10 vulnerability

OWASP TOP 10

- Injection (SQL, NoSQL, OS, and LDAP)
- Broken Authentication
- Sensitive Data Exposure
- XML External Entities (XXE)
- Broken Access Control
- Security Misconfiguration
- Cross-Site Scripting XSS.
- Insecure Deserialization
- Using Components with Known Vulnerabilities
- Insufficient Logging & Monitoring

https://www.owasp.org/index.php/Top_10_2013-Top_10

SQL INJECTION

- Tecnica usata per attaccare applicazioni di gestione dati
- Sia applicazioni Web che desktop
- Inietta codice SQL malevolo
- Query costruita dal programma:
`"SELECT * FROM users WHERE name = '" + userName + "'"`
- Parametri:
`userName = "' OR '1'='1 "`

demo time



NON SOLO SQL

- HTML injection
- Code injection (eval, scripting engine)
- XPath
- Nomi di file
- Accesso alle risorse
- Nomi di processi

SQL INJECTION

- Uso dei parametri nelle query
- Quotare le stringhe (QuotedStr)
- QueryBuilder
- Usare le espressioni regolari
- MAI usare direttamente l'input dell'utente

REGEX

- Class shorthands
- Brackets, Ranges and Negation `[]` , `-` and `^`
- Search Positioning (aka Anchors) `^` and `$`
- Iteration (aka Quantifiers) `?` , `*` , `+` , `{n}` , `{n.m}` and `{n,}`
- Parenthesis and Alternation `()` and `|`

Online regex tester <https://regex101.com/>

WIRL?

→ Validatori

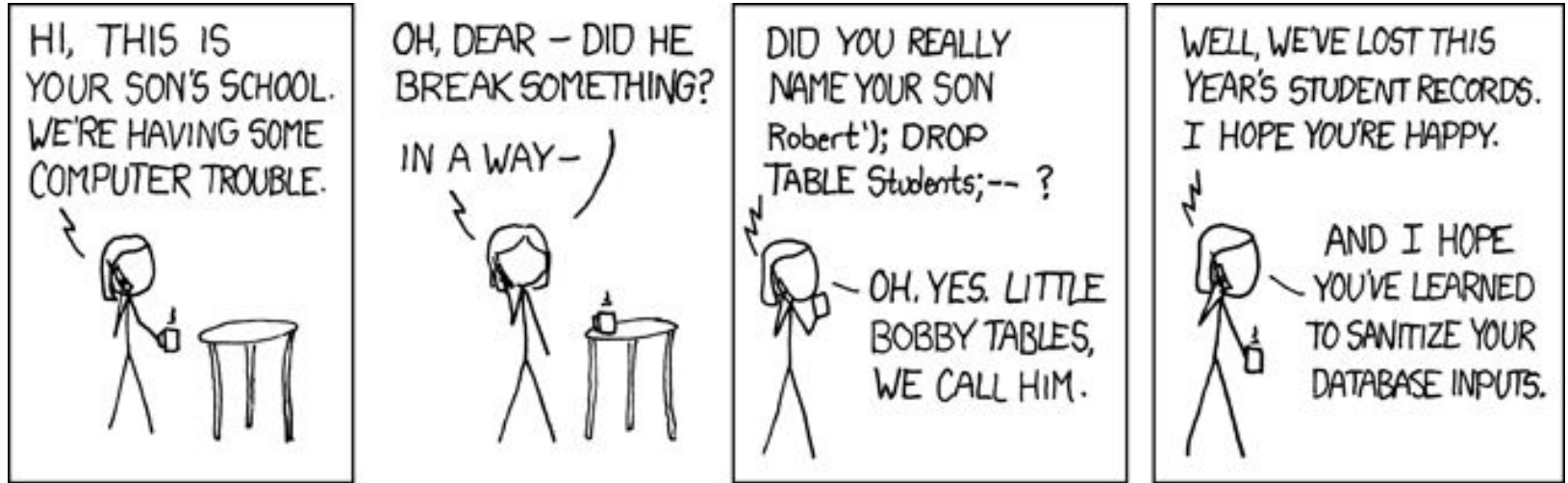
- ◆ Built-in: Max, Min, Pattern, NotNull, Size
- ◆ Custom
 - Standard / RawConstraint
 - Inizializzazione (Initialize)
 - Verifica (IsValid)

Query builder

GpSQLBuilder di Primož Gabrijelčič
<https://github.com/gabr42/GpSQLBuilder>

```
query := CreateGpSQLBuilder
    .Select
        .Column(['DISTINCT', DBPLY_DBField_ID])
        .&Case
            .When([DBSBF_DBField_CreatedTime, '< '2010-01-01'])
                .&Then('*')
            .&Else('')
        .&End
    .From(DBT_PLY)
    .LeftJoin(DBT_SBF)
        .&On([DBSBF_PLY_ID, '=', DBPLY_DBField_ID])
    .OrderBy(DBPLY_DBField_ID);
```

BOBBY TABLES?



<https://xkcd.com/327/>

XSS

- Tecnica che permette di inserire in una pagina web del codice malevolo
- Il codice inserito sarà eseguito da altri utenti e usato per carpire dati riservati
- Spesso viene recuperato l'ID di sessione i inviato a siti terzi

demo time



PASSWORD

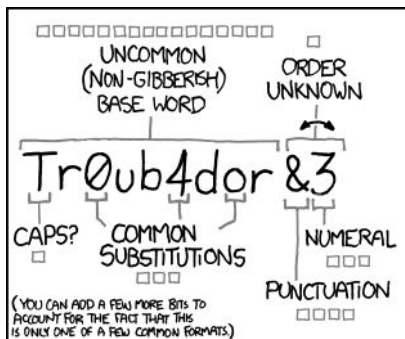
- Se è possibile accedere al DB leggere le password è il male minore (o no?)
 - ◆ Injection?
 - ◆ Gli utenti riciclano spesso le password
- Fornire agli utenti un mezzo per valutare la robustezza della propria password
- Cambiare spesso le password è una buona idea?

“

There's no question that the state of password security is problematic and has been for a long time. When humans pick their own passwords, too often they are easy to guess or predict. When humans are assigned or forced to create passwords that are hard to remember, too often they'll write them down where others can see them. When humans are forced to change their passwords, too often they'll make a small and predictable alteration to their existing passwords, and/or forget their new passwords. When passwords or their corresponding hashes are stolen, it can be difficult at best to detect or restrict their unauthorized use.

Recent scientific research calls into question the value of many long-standing password-security practices such as password expiration policies, and points instead to better alternatives such as enforcing banned-password lists (a great example being [Azure AD password protection](#)) and multi-factor authentication. While we recommend these alternatives, they cannot be expressed or enforced with our recommended security configuration baselines, which are built on Windows' built-in Group Policy settings and cannot include customer-specific values.

PASSWORD



~28 BITS OF ENTROPY

$2^{28} = 3 \text{ DAYS AT } 1000 \text{ GUESSES/SEC}$

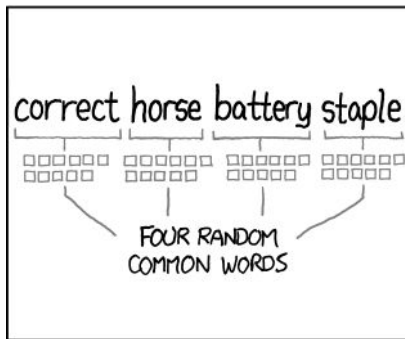
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)

DIFFICULTY TO GUESS: EASY

WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE 0s WAS A ZERO?

AND THERE WAS SOME SYMBOL...

DIFFICULTY TO REMEMBER: HARD



~44 BITS OF ENTROPY

$2^{44} = 550 \text{ YEARS AT } 1000 \text{ GUESSES/SEC}$

DIFFICULTY TO GUESS: HARD

THAT'S A BATTERY STAPLE.

CORRECT!

DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT

THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

PASSWORD

- Non inventare un proprio schema di cifratura
- Codificare la password con un algoritmo di HASH (possibilmente lento)
- Usare un salt per evitare attacchi di tipo “rainbow table attack”

PASSWORD

- MD5 e SHA sono progettati per essere veloci
- Usare preferibilmente
 - ◆ PBKDF2
 - ◆ Bcrypt
- Entrambi permettono di definire un “Work factor”. Essenzialmente il numero di iterazioni eseguite

PASSWORD

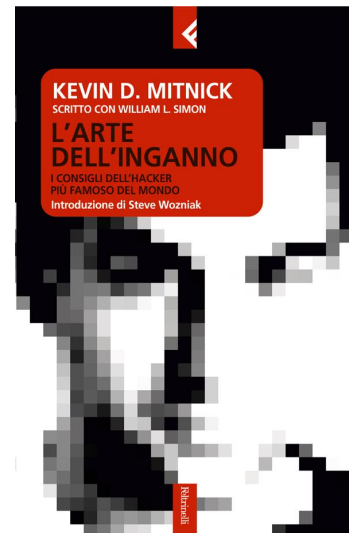
- Creazione dell'utente da parte dell'admin
 - ◆ Nessuno dovrebbe conoscere la password dell'utente (nemmeno il supporto tecnico)
 - ◆ Per applicazioni web l'admin dovrebbe inviare un link ad una pagina di primo login
 - ◆ Per applicazioni desktop dovrebbe fornire una password valida solo per il primo login
- L'utente deve essere autonomo nel ripristino della password (e-mail o admin)

demo time

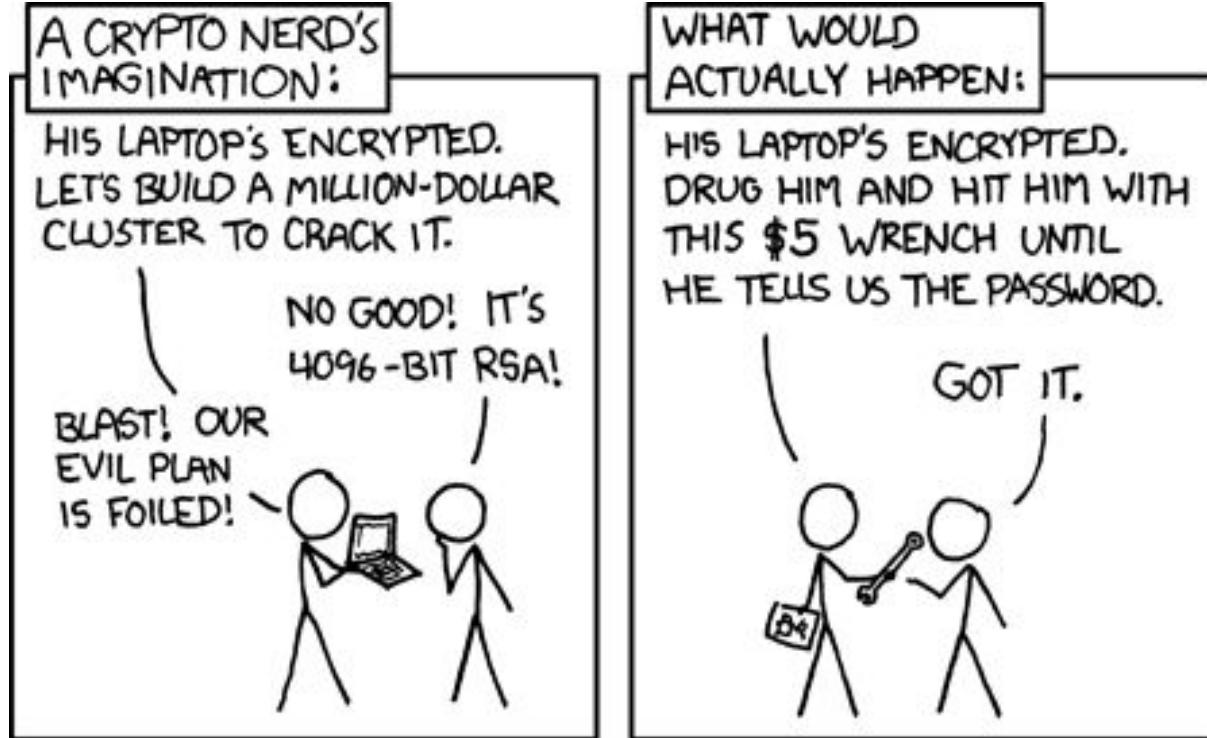


SOCIAL ENGINEERING

- “Pronto, sono del supporto tecnico. Mi può dare le sue credenziali per dei test?”
- Non abituare gli utenti a fornire le proprie credenziali
- Se avete bisogno di accedere al DB create un utente supervisore concordato col cliente



SOCIAL ENGINEERING





delphiedintorni.it



 wintech italia

THANK YOU



 wintech italia