Esercizio W10D1 - Simulazione Fase di Raccolta

RICHIESTA

L'esercizio W10D1 verte sull'utilizzo dei comandi di Google Hacking, al fine di raccogliere informazioni su un sito web. Nello specifico, l'esercizio richiede di provare quattro comandi (site: - inurl: - intext: - filetype:) e documentare i risultati ottenuti. Fra gli obiettivi principali, la ricerca di eventuali vulnerabilità o l'esposizione indebita di informazioni personali.

~~~

SOLUZIONE

Per la risoluzione dell'esercizio ho utilizzato i comandi di Google Hacking sul sito **www.dday.it**. Con i quattro comandi indicati dalla traccia, ho controllato prima l'intera struttura del sito, poi in quali altre pagine comparisse il sito dday.it (importante perché ha trovato pagine ufficiali di Facebook e LinkedIn, che ampliano la superficie del possibile attacco). Infine ho utilizzato i comandi per cercare testo all'interno della pagina (con l'intenzione di cercare eventuali riferimenti ad admin e password) e per scansionare il sito alla ricerca di un particolare formato di file (ho cercato un file di testo confidando sul fatto che un webmaster disattento o non particolarmente capace salvasse le password in un file del genere).

Le immagini successive mostrano come i comandi di Google Hacking abbiano funzionato ma che, al tempo stesso, da questa prima sommaria raccolta di informazioni non è spuntata ancora una possibile vulnerabilità. La "scoperta" delle pagine ufficiali di DDay.it su Facebook e LinkedIn potrebbe però rappresentare un elemento sul cui indagare alla ricerca di vulnerabilità.

~~~

ANNOTAZIONI TECNICHE

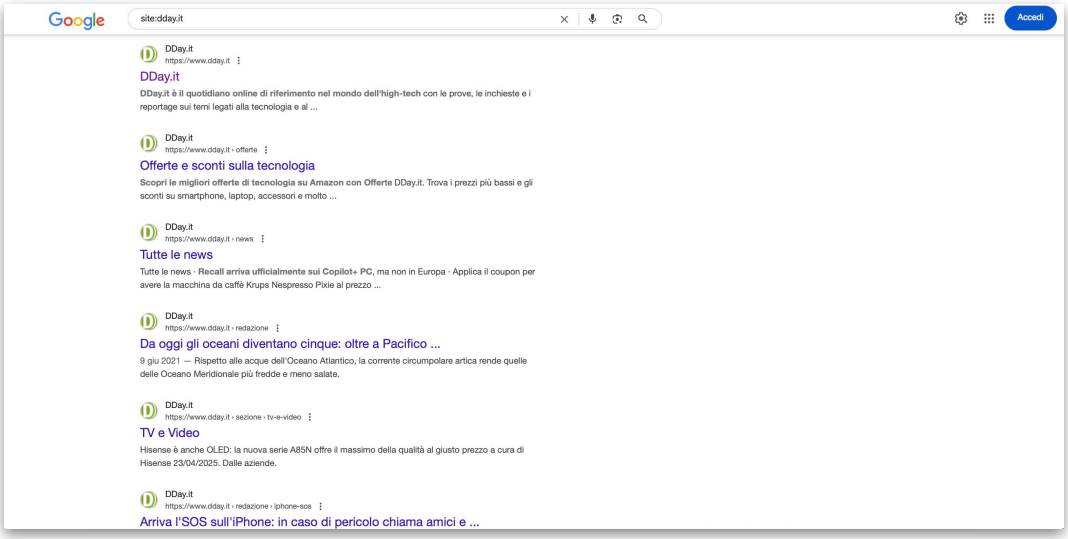
Il Google Hacking rappresenta una metodologia di information gathering che sfrutta avanzate funzionalità di ricerca del motore di Google. In particolare, dietro questa tecnica si nasconde la possibilità di utilizzare operatori di ricerca specifici, così da **affinare le query di ricerca** e individuare informazioni altrimenti non facilmente accessibili tramite ricerche convenzionali.

Il Google Hacking è detto anche Google Dorking, e fra le sue prerogative ha anche quella di filtrare i risultati in base a criteri impostati dall'utente in fase di ricerca. Questa tipologia di hacking è molto sfruttato nell'ambito della cybersecurity, sia in fase difensiva (identificazione di vulnerabilità, configurazioni errate e altro ancora) che in fase offensiva (raccolta di informazioni e scoperta di vulnerabilità).

Il target da me scelto per questa esercitazione è **www.dday.it**, portale di informazione sull'hi-tech di proprietà della società Scripta Manent servizi editoriali S.R.L.

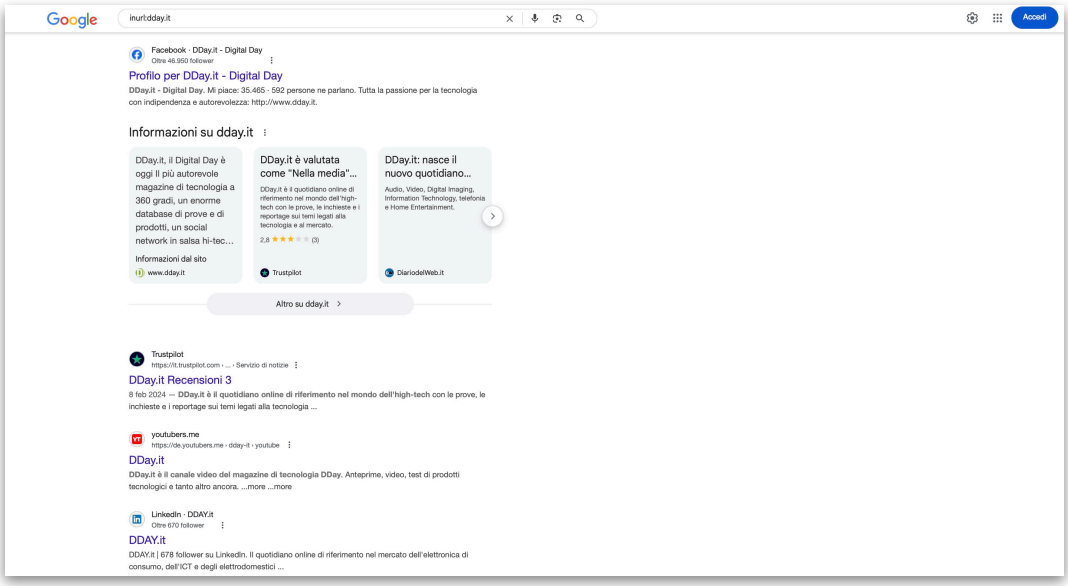
Il primo dei comandi di Google Hacking che ho provato è **site:dday.it**. Questo operatore restringe i risultati della ricerca alle sole pagine web che appartengono al dominio specificato, mostrando in pratica la struttura dell'intero sito web.

Nel nostro caso, si può notare come la struttura di DDay sia composta da una homepage, di una parte relativa agli articoli scritti (con data direttamente nell'URL) e da varie sezioni tematiche (Offerte, TV e Video etc.) tutte afferenti alla Homepage.



Il secondo comando che ho provato è **inurl:dday.it**. Il suo scopo è evidenziare quali pagine web contengano la stringa specificata (nel nostro caso dday.it). Guardando ai risultati (immagine successiva), si nota come la ricerca restituisca l'occorrenza dday.it anche in altre pagine, diverse dal sito stesso.

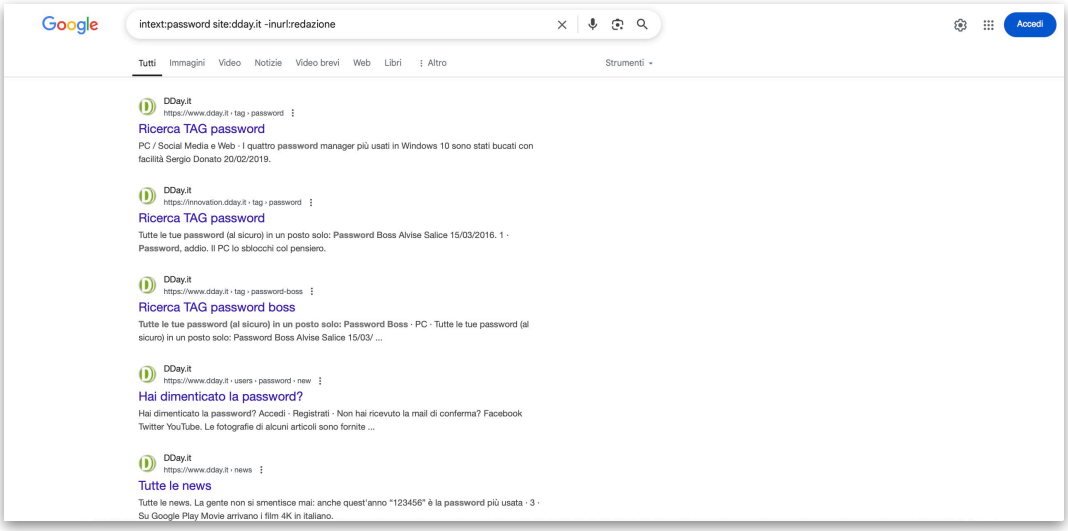
In particolare possiamo notare come dday.it compaia in Facebook (pagina ufficiale del portale), in Trustpilot (recensioni lasciate da utenti), in YouTube (canale ufficiale del portale) e nel feed di notizie proposte da Google. La differenza rispetto a site: è netta, visto che i risultati ottenuti col primo comando limitavano la ricerca solo al sito in questione.



Il terzo comando che ho provato è **intext:**, il quale permette di cercare pagine web all'interno del dominio specificato che contengano un determinato testo. Ho utilizzato questo comando con due diverse parole chiave: 'password' e 'admin'. Entrambe le ricerche erano volte a individuare eventuali pagine di

amministrazione facilmente accessibili o potenziali vulnerabilità legate alla gestione delle password degli utenti

In entrambi i casi la ricerca non ha evidenziati criticità. La ricerca con "password" ha restituito un gran numero di articoli scritti sul tema, - coerentemente con l'attività del portale - e lo stesso succede con admin. In base a questa ricerca, si può dire che il sito in questione non esponga sezioni sensibili del sito.
Tuttavia, data la natura informativa del sito e la potenziale presenza di articoli dedicati a questi argomenti (solitamente contenuti nella sezione 'redazione'), ho voluto affinare ulteriormente la ricerca escludendo questa sezione dagli URL. Anche in questo caso, la ricerca non ha rivelato alcuna falla di sicurezza.



Per concludere, ho poi provato il comando **filetype:txt site:dday.it**, cercando file .TXT. La mia ipotesi è che un webmaster distratto o incompetente possa aver conservato le password degli utenti in un file di testo, in chiaro. Salvare le password in chiaro è una procedura fortemente sconsigliata, ma ancora in uso.

Come si può vedere dall'immagine, la ricerca di Google evidenzia che non ci sono risultati. Questo vuol dire che non c'è nessun file .TXT pubblico su DDay.it. Si può ipotizzare, alla luce di questa prima indagine, che non ci sono vulnerabilità legate alla conservazione delle password sul sito.