# TORO Assicurazioni

## Allegato tecnico di risposta alla richiesta di offerta del 24 Novembre 2005

## *Attività di analisi della sicurezza della rete del Gruppo Toro Assicurazioni*

Milano


| | |
|---|---|
| **Hacking Team S.r.l.** | http://www.hackingteam.it |
| *Via della Moscova, 13*<br>*20121 MILANO (MI) - Italy* | info@hackingteam.it |
| *Tel. +39.02.29060603* | *Fax +39.02.63118946* |

| STORIA DEL DOCUMENTO | | |
|---|---|---|
| Versione | Data | Modifiche Effettuate |
| 1.0 | 13 Dicembre 2005 | Emissione |
| 1.1 | 6 Aprile 2006 | Sostituzione attività come da richiesta |
| // | // | // |

| INFORMAZIONI | |
|---|---|
| Data di Emissione | 13 Dicembre 2005 |
| Versione | 1.1 |
| Tipologia Documento | Allegato tecnico |
| Numero di Protocollo | // |
| Numero Pagine | 48 |
| Numero Allegati | 2 |
| Descrizione Allegati | Appendice 1 - Ethical Hacking.pdf<br>Report anonimizzato consegnato ad un cliente e riguardante attività di ethical hacking |
| | Appendice 2 - Analisi applicativa.pdf<br>Report anonimizzato consegnato ad un cliente e riguardante attività application assessment |
| Redatto da | Gianluca Vadruccio |
| Approvato da | Valeriano Bedeschi |

# INDICE

# 1 Obiettivo

Lo scopo del presente documento è quello di rispondere alla richiesta di offerta tecnica ed economica emessa da TORO Assicurazioni in data 24 Novembre 2005 attraverso l'invio di una e-mail.

Verrà risposto, punto per punto, a tutto quello richiesto nella documentazione inviata, cercando di mantenere lo stesso ordine di richiesta e le stesse terminologie.

L'oggetto della richiesta riguarda in generale l'erogazione di attività di analisi della sicurezza della rete del Gruppo Toro Assicurazioni ed in particolare:

- ***target***: penetration test sia interno che esterno, sia di rete che applicativo, analisi della sicurezza della VPN, wardialing, wardriving e definizione di un team di incident handling
- ***modalità***: è richiesto il penetration test sia in modalità black-box dall'esterno (senza alcuna credenziale), sia in modalità white-box dall'interno (differenziando credenziali utente e credenziali amministrative)
- ***time***: deve essere specificato, per ogni attività, l'effort richiesto, i tempo di attivazione e la stima dei tempi

Il documento descriverà la metodologia che si intende seguire nei vari casi di assessment, il team assegnato alle varie attività, la struttura e le esperienze del team di security engineer proposto. Inoltre si evidenzieranno tutte quelle informazioni espressamente richieste nel documento di specifiche tecniche.

# 2 Composizione del team di Security Assessment

## *2.1 L'organizzazione del settore security*

La parte security dell'offerente possiede il seguente portafoglio di offerta:

1. Servizi di assessment: penetration test, vulnerability assessment, ethical hacking, social engineering, application assessment, security plan, fixing strategy ovvero, in generale, studi di sicurezza in ambito ICT.
2. Soluzioni di security: delivery di soluzioni ad hoc soddisfacenti le esigenze ed i requisiti del cliente, anche attraverso lo sviluppo di software e personalizzazioni molto spinte.
3. TroubleShooting: individuazione di malfunzionamenti, risoluzione di problematiche di network e security, application stress test analysis, code review

4. System Integration: progettazione, personalizzazione e delivery di soluzioni di sicurezza basate su prodotti di terze parti. A puro titolo di esempio, elenchiamo sicurezza perimetrale, identity management e provisioning, log management e event correlation, desktop security e policy enforcement, strong authentication e single sign-on…

Il personale di security consultant è strutturato in tre aree:

1. Hackers Team: personale focalizzato su security assessment ed ethical hacking
2. Developers Team: personale maggiormente incline allo sviluppo di security software e security customization
3. Projects Team: personale con esperienza in progettazione di soluzioni di sicurezza ed integrazione di sistemi

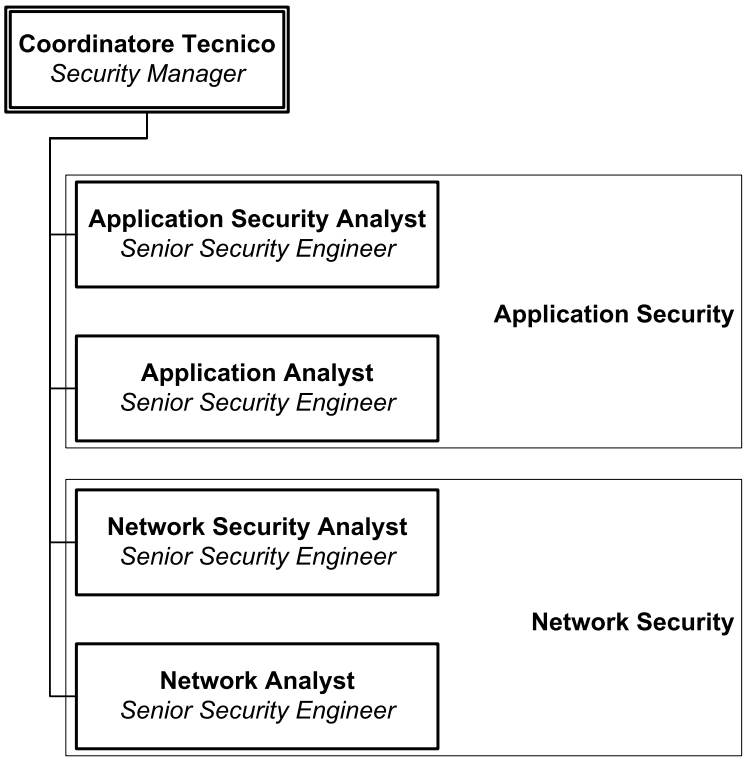
Solitamente, il team dedicato al soddisfacimento di un cliente viene composto da figure selezionate ad hoc all'interno di queste aree. L'unione di queste competenze complementari, fornisce al delivery caratteristiche di qualificata progettazione, elevato livello di sicurezza ed enormi potenzialità di personalizzazione.

## *2.2 Il team di assessment*

Il team che verrà costituito in caso di assegnazione positiva della gara è di notevole caratura tecnica e professionale ed è costituito da personale tecnico operante nel settore della sicurezza informatica da più di dieci anni.

L'importante esperienza maturata sul campo negli anni (vedere il paragrafo 3), permette di poter annoverare nella squadra elementi di altissima qualità in tutti gli ambiti richiesti dal cliente: vulnerability assessment, penetration test e application assessment.

Il team proposto è schematizzato nella seguente figura.




Ai tecnici specifici d'area è affiancato un valido analizzatore di sistemi della stessa area, in maniera tale da garantire supporto di conoscenza anche al di fuori della pura security. Quindi, ad esempio, alla persona incaricata degli assessment in ambienti applicativi verrà affiancata all'occorrenza una persona esperta del settore applicativo. Questo garantisce una analisi completa ed esaustiva della tematica in oggetto.

Qualora venisse inserito anche l'assessment sistemistico, al team si potrà aggiungere la figura di System Security Analyst, avente anche lui la qualifica di Senior Security Engineer.

# 3 Esperienze di assessment

Nel presente paragrafo verranno fornite le principali esperienze dell'offerente effettuate nell'ultimo anno esclusivamente nell'ambito dei security assessment. Per evidenti ragioni, si ometteranno i nomi delle società; si evidenzieranno le tipologie di attività svolte ed il relativo dimensionamento. Ad ogni esperienza è dedicata una tabella riassuntiva.

| **Tipologia cliente: Assicurazione di media dimensione** | |
|---|---|
| Tipologia di attività | Assessment applicativo ed analisi della |

| | |
|---|---|
| | sicurezza wireless |
| Breve descrizione | Analisi della sicurezza di tutte le applicazioni interne ed esterne e studio del sistema e delle configurazioni wireless esistenti |
| Dimensionamento | 15 IP e 3 applicazioni |

| **Tipologia cliente: Assicurazione di grande dimensione** | |
|---|---|
| Tipologia di attività | Assessment esterno completo |
| Breve descrizione | Analisi della sicurezza di tutte le applicazioni esterne e penetration test del perimetro |
| Dimensionamento | 16 IP e 9 portali applicativi |

| **Tipologia cliente: Banca di grande dimensione** | |
|---|---|
| Tipologia di attività | Assessment applicativo |
| Breve descrizione | Analisi della sicurezza delle applicazioni esterne |
| Dimensionamento | 2 portali applicativi + 1 portale di home-banking |

| **Tipologia cliente: Assicurazione di media dimensione** | |
|---|---|
| Tipologia di attività | Assessment interno ed esterno completo |
| Breve descrizione | Analisi della sicurezza di tutte le applicazioni esterne e penetration test del perimetro |
| Dimensionamento | 50 IP e 3 applicazioni |

| **Tipologia cliente: Industria di media dimensione** | |
|---|---|
| Tipologia di attività | Penetration test |
| Breve descrizione | Penetration test dall'esterno con azioni spinte di exploiting |
| Dimensionamento | 16 IP |

| **Tipologia cliente: Assicurazione di media dimensione** | |
|---|---|
| Tipologia di attività | Penetration test |
| Breve descrizione | Penetration test dall'esterno con azioni spinte di exploiting |
| Dimensionamento | 16 IP |

| **Tipologia cliente: Industria di grande dimensione** |
|---|

| | |
|---|---|
| Tipologia di attività | Analisi sicurezza sistema RAS |
| Breve descrizione | Studio del sistema di accesso remoto ed analisi delle potenziali aree migliorative |
| Dimensionamento | RAS utilizzato da 1.200 utenze |

| **Tipologia cliente: Banca di media dimensione** | |
|---|---|
| Tipologia di attività | Analisi sicurezza sistema transazionale |
| Breve descrizione | Verifica del sistema di sicurezza transazionale legato all'utilizzo delle carte di credito via POS e via Internet |
| Dimensionamento | 30 IP e 1 portale applicativo |

| **Tipologia cliente: Utilities di media dimensione** | |
|---|---|
| Tipologia di attività | TroubleShooting and System analysis |
| Breve descrizione | Identificazione di problematiche legate al traffico di rete e al sistema di bilanciamento ed alta affidabilità |
| Dimensionamento | 10 IP e 1 applicazione |

| **Tipologia cliente: Assicurazione di media dimensione** | |
|---|---|
| Tipologia di attività | Application stress test analysis |
| Breve descrizione | Rilevamento di problematiche legate all'utilizzo concorrente di due applicazioni ed individuazione dei colli di bottiglia |
| Dimensionamento | 2 applicazioni |

| **Tipologia cliente: Militare di media dimensione** | |
|---|---|
| Tipologia di attività | Penetration test |
| Breve descrizione | Penetration test dall'esterno con azioni spinte di exploiting |
| Dimensionamento | 16 IP |

| **Tipologia cliente: Banca di media dimensione** | |
|---|---|
| Tipologia di attività | Assessment interno ed esterno completo |
| Breve descrizione | Analisi della sicurezza di tutte le applicazioni esterne e penetration test del perimetro |
| Dimensionamento | 16 IP e 4 portali applicativi e 1 portale di home-banking |

| Tipologia cliente: Banca di grande dimensione | |
|---|---|
| Tipologia di attività | Vulnerability assessment interno completo |
| Breve descrizione | Analisi della sicurezza della rete interna e dei servizi erogati internamente |
| Dimensionamento | 50 IP |

| Tipologia cliente: Istituto finanziario di media dimensione | |
|---|---|
| Tipologia di attività | Studio di sicurezza sul sistema di autenticazione |
| Breve descrizione | Analisi della sicurezza del sistema di strong authentication esistente e identificazione di una strategia migliorativa |
| Dimensionamento | 1 portale applicativo |

| Tipologia cliente: Istituto governativo di grande dimensione | |
|---|---|
| Tipologia di attività | Consulenza di sicurezza ed analisi forense |
| Breve descrizione | Attività consulenziale in ambito di assessment sia di rete che applicativo ed attività di intelligence molto avanzata |
| Dimensionamento | // |

| Tipologia cliente: Assicurazione di grande dimensione | |
|---|---|
| Tipologia di attività | Attività di sicurezza su tutta l'infrastruttura tecnologica |
| Breve descrizione | ○ Penetration test di rete e applicativo sul perimetro esterno<br>○ Vulnerability assessment di rete ed applicativo interno<br>○ Vulnerability assessment di tutto il mondo delle agenzie e subagenzie<br>○ Analisi forense su server specifici di produzione |
| Dimensionamento | 300 IP, 2 portali applicativi, 30 applicazioni |

| Tipologia cliente: Servizi bancari di grande dimensione | |
|---|---|
| Tipologia di attività | Analisi dei sistemi transazionali di produzione |
| Breve descrizione | Analisi della configurazione, della profilatura e del controllo accessi di server critici |
| Dimensionamento | 20 IP |

| Tipologia cliente: Servizi di media dimensione | |
|---|---|
| Tipologia di attività | Assessment applicativo |
| Breve descrizione | Analisi delle vulnerabilità applicative del portale critico |
| Dimensionamento | 1 portale applicativo |

| Tipologia cliente: Servizi di media dimensione | |
|---|---|
| Tipologia di attività | Assessment applicativo |
| Breve descrizione | Analisi delle vulnerabilità applicative del portale critico |
| Dimensionamento | 1 portale applicativo |

| Tipologia cliente: Banca di grande dimensione | |
|---|---|
| Tipologia di attività | Assessment applicativo |
| Breve descrizione | Analisi della sicurezza del portale principale |
| Dimensionamento | 1 portale di home-banking |

# 4 Referenze, articoli e pubblicazioni

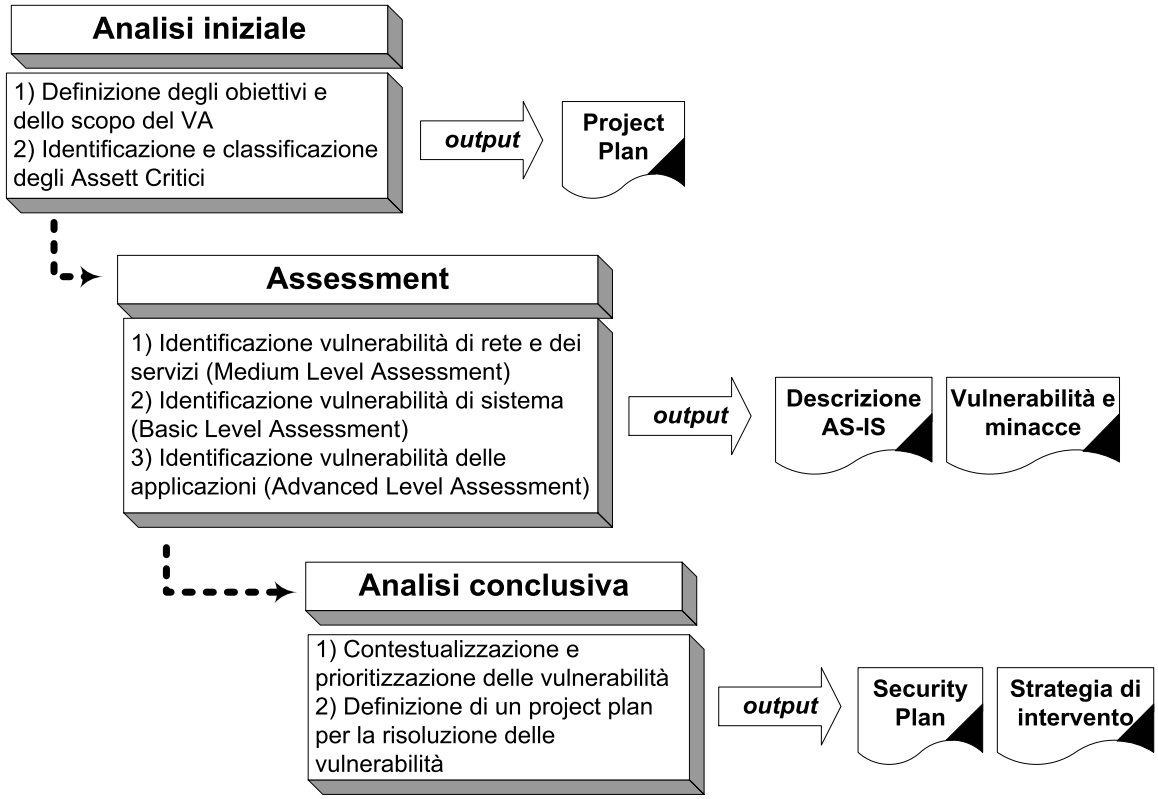
Si elencano di seguito gli eventi principali di security che ha visto coinvolto l'offerente nei due anni passati:

- Articolo 2003 - Consorzio per l'innovazione tecnologica - Università di Brescia (http://www.inntec.it/magazine/N44.html)
  COME VERIFICARE LA SICUREZZA DEL PROPRIO SISTEMA INFORMATIVO
- Articolo 2005 - (IN)Sicurezza di Bluetooth
  Relatore a convegni/seminari IRI, Systech, LRA, Bocconi, Infosecurity
- Speech:
  - speaker al convegno Cisco EMEA 2003
  - speaker a conferenza BlackHat europe 2003
  - speaker a conferenza BlackHat USA 2003
  - ethical hacker speech II (blackhats.it) a SMAU 2002
  - speaker al convegno Webbit 2002/2003/2004
  - attack demos a Infosecurity 2002
  - ethical hacking - Infosecurity 2001

  - ethical hacking - FORUM PA 2002 Roma
- Interventi e docenze per:
  - Università Milano Statale
  - Università Milano Bocconi
  - Università di Firenze
  - CNR
  - CLUSIT
  - LRA (insieme a V.Bedeschi)
- Software, paper, presentazioni, misc:
  - ettercap (strumento di attacco Man in the Middle)
  - MITM Attacks (paper per blackhats.it e presentazione portata a cisco emea e alle due conferenze blackhat USA)
  - Advanced windows exploiting I (per BFi, rivista underground)
  - Advanced windows exploiting II (per webbit, convegno)
  - VPN Attacks (per webbit)
  - e-gold dove sono i miei soldi? (a proposito delle carte di credito, pagamenti su internet, paypal, etc. per webbit, convegno)
  - beholder (per BFi, rivista underground)
  - PE manipulation (per ringzer0, rivista underground)
- Evento Security Date 2004 - Steganografia, l'arte della scrittura nascosta
- Pubblicazione underground BFi
  - articolo BFi#12 "steganografia su sessioni di rete e dintorni"
  - articolo BFi#13 "SABBIA sulla bassa latenza"
  - paper BFi#12 "Advanced Windows Exploiting"

# 5 Descrizione dell'approccio di assessment

Hacking Team ha adottato una metodologia di "Vulnerability Assessment" che prevede tre distinte fasi, schematizzate nella figura seguente: **Analisi Iniziale, Assessment e Analisi Conclusiva**.




## *5.1 Analisi Iniziale*

L'analisi iniziale è la prima fase che caratterizza un assessment e consiste nel definire con precisione gli obiettivi di tale attività. Come per ogni servizio deve essere chiaro sia al committente, sia al mandatario, quali sono i risultati attesi dalla prestazione fornita. A tale scopo è importante definire e pianificare con precisione diversi aspetti:

- **L'oggetto dell'assessment**: è necessario identificare i confini entro cui condurre l'attività in termini di sistemi informativi coinvolti.
- **Il contesto operativo**: è necessario definire la modalità e i tempi secondo cui sarà condotta l'attività sia in termini di strumenti, sia in termini di risorse coinvolte da entrambi le parti.
- **Il livello di accuratezza**: è necessario stabilire il dettaglio che si intende raggiungere con l'assessment al fine di pianificare correttamente le attività successive. Questo aspetto è in parte già chiarito nel paragrafo degli obiettivi del presente documento: il livello di approfondimento garantito è quello specificato nella Fase 1.

- **Deliverables**: è necessario definire a priori il modello e la tipologia di documentazione che l'assessment produrrà al fine di soddisfare le esigenze della parte committente. Solitamente la documentazione prodotta è la seguente:
    - Documento di progetto
    - Security Plan (piano di intervento e contromisure a copertura delle vulnerabilità)

Sulla base di questa analisi iniziale saranno pianificate le varie attività che caratterizzeranno le due fasi successive: assessment e analisi conclusiva. E' importante sottolineare che per valutare correttamente l'importanza delle vulnerabilità riscontrate durante la fase di assessment, i soli aspetti tecnici non sono sufficienti. Questo perché la vulnerabilità deve essere ponderata sulla base della criticità degli asset coinvolti. L'importanza di identificare gli asset critici è ancora più evidente nella definizione del piano di risoluzione, dove la priorità e la modalità degli interventi deve essere relazionata al valore dell'asset stesso.

## *5.2 Assessment*

Questa parte della metodologia è ampiamente dettagliata nei rispettivi paragrafi 6.1 Assessment sistemistico, 6.2 Assessment di rete e dei servizi, 6.3 Assessment applicativo.
Ovviamente a seconda dello scenario e del target che si presentano, l'assessment si occuperà di uno di questi tre livelli oppure di un misto dei tre nel caso di analisi complesse e diversificate.

## *5.3 Analisi conclusiva*

L'analisi conclusiva, come si evince dal termine stesso, ha lo scopo di concludere l'assessment, fornendo la documentazione contenente i log di evidenza dell'attività svolta, i report delle varie tipologie di vulnerabilità riscontrate e il piano d'intervento consigliato.
La parte di documentazione relativa ai report delle vulnerabilità sarà strutturata secondo la classificazione descritta precedentemente e pesata sulla base delle criticità definite durante l'analisi iniziale. La gravità di una vulnerabilità sarà quindi frutto sia del livello d'importanza in termini tecnici (pericolosità[1]), sia in termini di business aziendale (criticità[2]).

[1] Con pericolosità s'intende quanto la vulnerabilità in questione comporti la possibilità di compromissione dei parametri di disponibilità, riservatezza e integrità dell'asset. Ad esempio, un *denial of service* è sicuramente meno pericoloso di un'escalation a diritti di amministrazione del sistema di un server critico.
[2] Con criticità s'intende quanto la vulnerabilità in questione possa avere un impatto sul business aziendale. Ad esempio, un exploit remoto sarà più grave se legato a un server dove è presente il database dei clienti, rispetto al PC della segreteria non ospitante dati sensibili.

Il security plan ed il piano d'intervento saranno, in modo analogo alla classificazione delle vulnerabilità, redatti tenendo in considerazione la gravità delle lacune riscontrate. Il Security Plan, che contiene il piano di intervento, è invece di più ad alto livello e comprende la strategia di sicurezza che il cliente porterà avanti con l'intento di aumentare il livello di sicurezza globale (reti, sistemi, applicazioni e procedure).

# 6 Metodologia seguita negli assessment

## *6.1 Assessment sistemistico*

Questo genere di assessment richiede una notevole esperienza in ambito sistemistico; infatti le figure professionali che vengono coinvolte sono esperti conoscitori dei sistemi e delle piattaforme, abituati a focalizzare maggiormente l'attenzione sugli aspetti di sicurezza.
In un assessment completo è importante tenere in considerazione anche questa tipologia, che evidenzia vulnerabilità e minacce presenti, anche se non *raggiungibili* dal perimetro esterno. In questo caso (che costituisce solo un esempio di una moltitudine di casistiche possibili), l'attacco potrebbe essere portato a compimento dall'interno, cioè dalla parte in cui la vulnerabilità si *rende visibile*.
Le principali aree tematiche da tenere in considerazione durante un'analisi di sicurezza sistemistica sono evidenziate dai seguenti passi, mostrati anche sinteticamente nella figura successiva.

***Configurazione generale***

Ci si occupa della piattaforma in generale, partendo dai parametri di configurazione di sistema, dei dispositivi, dei files principali e di conf, ed arrivando alle variabili di ambiente e dei registri (in caso di sistema operativo microsoft). Si terranno in considerazione, come linee guida, le *best practices* relative alla piattaforma in oggetto e lo stato dell'arte della sicurezza al momento dell'esecuzione dell'assessment.

***Accounting***

Il focus della presente tematica riguarda gli utilizzatori locali e remoti della piattaforma esaminata: si controlleranno le configurazioni e le profilature assegnate al parco utenti, al parco amministratori e a qualsiasi altro profilo esistente sulla macchina. Verranno analizzate la procedura e gli strumenti

di autenticazione ed eventuali scostamenti rispetto alle politiche scritte o definite e volute dal cliente.

***Amministrazione***

Con l'ausilio di brevi interviste e di verifiche sul campo, si procederà a rilevare le modalità e gli strumenti di amministrazione della piattaforma esaminata, nell'ottica di evidenziare vulnerabilità di tipo procedurale oppure vulnerabilità legate alla tipologia di gestione scelta. Ovviamente questo tipo di studio verrà eseguito sia per l'amministrazione locale sia per l'eventuale amministrazione remota.

***Integrità del sistema***

In questa parte si verifica lo stato di sicurezza del file system dal punto di vista della sua integrità, ricorrendo ad analisi locali specializzate che, in azioni di assessment ripetute, possono coincidere anche con gap analysis rispetto all'ultima azione effettuata. Qualora il cliente abbia una politica che definisca una base integra di partenza, la verifica dell'integrità potrebbe essere condotta in riferimento a quanto stabilito e voluto per quella determinata macchina.

***Log analysis***

Per particolari tipologie di assessment di tipo basic, potrebbe esserci la necessità di risalire ad un particolare evento passato oppure di rilevare una particolare situazione anomala. Questo è il caso in cui si ricorre all'analisi dei log di sistema. Qualora questa attività non sia necessaria o non sia richiesta, ci si limiterà a verificare che la configurazione e la gestione del sistema di logging sia aderente a quanto definito.

***Politiche***

Questa parte riguarda la presenza di eventuali politiche definite dal cliente su tutto iul parco macchine e/o sulla machina specificatamente analizzata. Si controllerà insieme al cliente l'effettiva aderenza a quanto da lui stesso voluto, verificando ad esempio la procedura di backup/restore della configurazione, il salvataggio e la protezione di eventuali dati sensibili, …

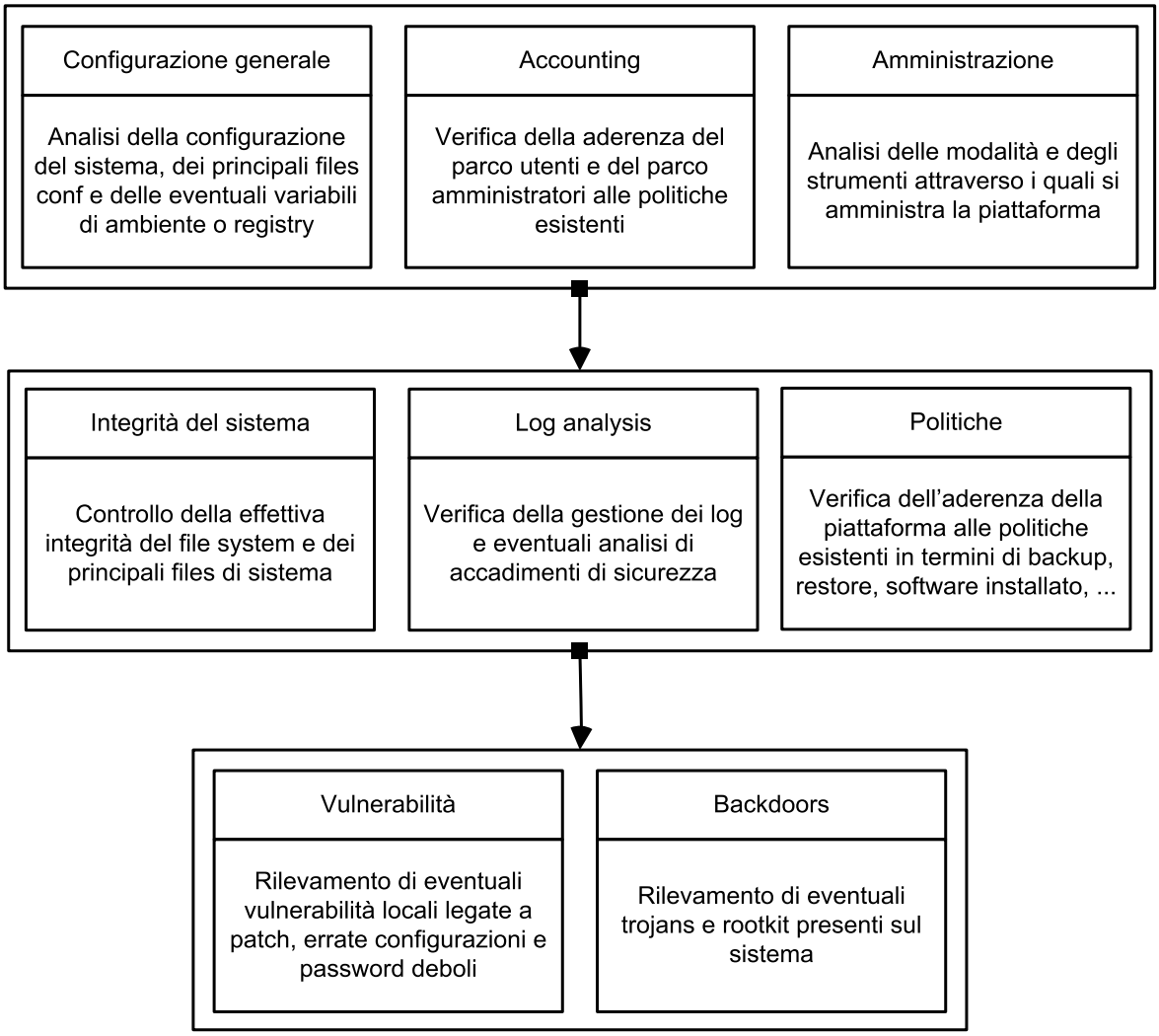
E' questa la fase in cui si identificano anche installazioni estranee o non autorizzate rispetto a quanto stabilito esserci sulla macchina di riferimento.

***Vulnerabilità***

Si esegue il rilevamento di vulnerabilità esistenti localmente e relative ad errate configurazioni oppure alla necessità di patch evolutive o di sicurezza. E' questa la parte in cui si verificherà inoltre la presenza di eventuali password deboli.

***Backdoors***

Un ruolo importante è assunto da questa fase, che consiste nel rilevamento di eventuali troiani o rootkit presenti sulla macchina ed esistenti a fronte di un attacco avvenuto in passato. La loro individuazione e rimozione è fondamentale per negare al possibile attaccante la possibilità e l'enorme vantaggio di avere una backdoor in suo favore.



## 6.2 Assessment di rete e dei servizi

Le attività di Ethical Hacking (vulnerability assessment e penetration test) da noi eseguite tentano di emulare al 100% il comportamento di un vero hacker.

***Analisi non invasiva***

1. FOOTPRINTING

   Questa fase ha lo scopo di raccogliere il maggior numero di informazioni sull'obiettivo che si intende attaccare senza "toccare" l'obiettivo stesso, ovvero effettuando una cosiddetta "analisi non invasiva". In particolare in questa fase si cerca di determinare: *domini, blocchi di rete e indirizzi IP dei sistemi direttamente collegati a Internet*. Gli strumenti utilizzati sono: Search Engine, Whois server, Arin database, interrogazione DNS, ecc.

2. SCANNING

   L'obiettivo dello scanning è ottenere una mappa più dettagliata possibile del sistema da attaccare. Ciò significa acquisire informazioni su quali indirizzi IP dei blocchi di rete trovati nella fase precedente siano effettivamente "contattabili" dall'esterno (IP discovery), quali servizi siano "attivi" (TCP/UDP port scan) e, infine, quali sistemi operativi "posseggano". Gli strumenti utilizzati sono: interrogazioni ICMP (gping, fping, ecc.), scansione delle porte TCP e UDP (strobe, netcat, nmap, ecc.), fingerprint dello stack (nmap, ethercap).

***Analisi invasiva***

3. ENUMERATION

   Con questa fase si inizia l'"analisi invasiva". Si effettuano, infatti, connessioni dirette ai server e "interrogazioni" esplicite. Tali attività potrebbero, a seconda della configurazione presente sui sistemi target, originare dei logs sui sistemi (tipicamente su sistemi di controllo). Attraverso l'enumerazione si vuole giungere a identificare, sulle macchine riscontrate come raggiungibili, account validi (list user accounts), risorse condivise (list file shares) e applicazioni attive sulle porte in ascolto (identify application). Le tecniche utilizzate variano a seconda dei sistemi operativi delle macchine che vogliamo analizzare.

***Attacco***

4. GAINING ACCESS

   Una volta ottenute le informazioni del punto precedente, inizia il vero e proprio attacco che ha come obiettivo riuscire a "entrare" nel sistema remoto. I metodi utilizzati anche in questo caso dipendono dal sistema operativo della macchina target, ma si basano sostanzialmente sulla

ricerca di password corrispondenti agli utenti trovati (password guessing), sullo sfruttamento di errori progettuali delle applicazioni e servizi attivi sul server (buffer overflows, attacchi data driven, ecc.) o del sistema operativo stesso.

5. ESCALATING PRIVILEGES[3]

   L'obiettivo di questa fase è sfruttare i risultati ottenuti nella fase precedente per ottenere il pieno controllo del sistema remoto attaccato. Ciò si ottiene, per esempio, reperendo i files presenti sul sistema che contengono le password (/etc/passwd, SAM, ecc.) e tentando di decifrare le password in essi contenute (password cracking), oppure utilizzando appositi exploits.

***Consolidamento***

6. PILFERING

   Se si giunge a questa fase significa che si è ottenuto il pieno controllo del sistema target. Quindi è bene valutare la configurazione del sistema stesso al fine di capire se, dove e cosa il sistema registra (logs). I sistemi di auditing saranno eventualmente disabilitati (es. con Win NT mediante auditpol). A questo punto la macchina in oggetto può diventare una "testa di ponte" per attaccare altre macchine. In tal caso saranno reperite informazioni riguardanti altri sistemi.

7. COVERING TRACES AND CREATING BACK DOORS

   Prima di abbandonare il sistema "conquistato" vengono cancellati gli eventuali logs che hanno registrato la presenza clandestina ed eventualmente installati trojan o back-doors che consentano di rientrare facilmente sulla macchina in un secondo momento. Può essere utile anche installare tools nascosti quali sniffers o keyloggers al fine di catturare altre password del sistema locale o di altri sistemi ai quali utenti ignari si collegano dalla macchina controllata.

## *6.3 Assessment applicativo*

Questa analisi è costituita da una serie di tentativi d'attacco che coinvolgono solo i protocolli di comunicazione utilizzati dagli utenti finali per interagire con le applicazioni.

Nel caso specifico delle applicazioni web, tali attacchi sono basati su manipolazioni dei pacchetti HTTP che vengono scambiati fra i browser degli utenti ed il web server. Esistono diverse categorie

[3] Vogliamo specificare che, considerata la natura della presente offerta, le nostre attività *non si spingeranno in nessun caso oltre questo punto (ESCALATING PRIVILEGES) a meno di una specifica autorizzazione in tal senso da parte del cliente.* In altre parole, si cercherà di **dimostrare l'effettiva possibilità di assumere il controllo dei sistemi senza apportare alcuna modifica agli stessi.**

di attacchi verso applicazioni web, che possono portare alla compromissione di uno o più layer dell'intera infrastruttura applicativa: web server, application server, data tier.
Caratteristica comune a tutti gli attacchi applicativi è la completa trasparenza ad ogni sistema di difesa perimetrale (firewall, ids, ecc.): manipolazioni dei protocolli di layer 7 (applicativi) non possono essere rilevate da dispostivi che analizzano il traffico a layer 3 (network).
Il test sarà condotto in modalità anonima ed in "user-mode". Ciò significa che, preventivamente, dovra' essere creato un account tramite le usuali procedure di attivazione al fine di permettere a Hacking Team di accedere come utente autorizzato. Non saranno accettati account di altro tipo (di test interno, amministrativi, etc.) poiché non fornirebbero la corretta valutazione circa il rischio che un utente registrato possa cercare di accedere in modo fraudolento ad informazioni per cui non è autorizzato. L'attività comprende l'analisi dell'applicazione in termini architetturali, verranno analizzate le configurazioni delle macchine interessate, sia a livello di sistema operativo che applicativo.
In generale, le vulnerabilità di livello applicativo sono spesso legate ad errori contenuti nel codice delle applicazioni. Esistono due classi di errori, che richiedono differenti strategie per essere identificati e rimossi: errori logico-architetturali ed errori di implementazione.

***Errori logico-architetturali***

Gli errori logico-architetturali consistono nel mancato utilizzo di meccanismi di sicurezza, oppure nell'utilizzo di meccanismi non adeguati a raggiungere lo scopo desiderato. Tali errori sono imputabili ad una non corretta definizione dei requisiti di sicurezza e/o ad una inadeguata progettazione dell'architettura.
Gli errori logico-architetturali più comuni sono i seguenti:

- gestione non corretta delle sessioni;
- uso di meccanismi di autenticazione deboli, che
  - permettono agli utenti di utilizzare password *guessable;*
  - rilasciano informazioni che permettono di restringere lo spazio di ricerca per attacchi di tipo *brute force*;
- trasmissione di informazioni sensibili su canali non cifrati;
- assunzioni errate in merito all'attendibilità e veridicità di input ricevuti dall'utente;
- assunzioni errate in merito alla funzionalità di sistemi e/o applicazioni *client-side* (ad esempio, browser web) che si trovano sotto il controllo dell'utente (o dell'attaccante!).

***Errori implementativi***

Questi errori si originano in fase di sviluppo, quando specifiche di alto livello, corrette dal punto di vista logico, vengono tradotte in codice che non gestisce correttamente tutti i casi i possibili; i malfunzionamenti che si verificano in casi particolari possono essere sfruttati per indurre nelle applicazioni comportamenti non previsti e/o non desiderati.

La grande maggioranza degli errori implementativi è dovuta ad una non corretta validazione dei parametri in ingresso, oppure alla gestione non corretta di alcuni input particolari, non previsti dal programmatore. La loro natura rende estremamente difficile prevederne l'impatto: in alcuni casi, questi errori possono avere conseguenze gravi sulla sicurezza di una applicazione, anche se gli elementi di codice interessati non sono direttamente legati a funzionalità critiche.

Gli attacchi di livello applicativo sfruttano vulnerabilità (sia di natura logico-architetturale, sia di natura implementativa) per indurre nelle applicazioni comportamenti anomali, le cui conseguenze possono essere le più disparate: crash dell'applicazione, furto di dati, accesso ai sistemi su cui le applicazioni sono eseguite, ecc.

Allo scopo di inquadrare il tema della sicurezza del livello applicativo, sia in termini di "opportunità" offerte all'intrusore, sia di minacce per le potenziali vittime di intrusioni, si dà una sintetica descrizione delle principali tecniche di attacco utilizzate nell'ambito delle applicazioni web.

I concetti e la terminologia introdotti saranno utilizzati nel presente documento per descrivere i risultati dell'assessment svolto.

### 6.3.1 Authentication brute-forcing

- ***Obiettivo***: accesso aree riservate ad utenti in possesso di opportune credenziali.
- ***Attaccanti***: chiunque non sia in possesso di credenziali valide ed abbia interesse ad accedere alle informazioni contenute nelle aree riservate, oppure chi, pur possedendo credenziali valide, intende accedere all'area riservata con l'identità di un altro utente.
- ***Descrizione***: consiste nella sottomissione, spesso con l'ausilio di tool automatici, di un grande numero di credenziali (ad esempio coppie username,password), fino ad ottenere una risposta di autenticazione riuscita dal sistema. La generazione delle credenziali può prevedere l'uso di regole (ad esempio, generazione di tutte la password di sei caratteri costituite da soli caratteri alfanumerici) oppure di dizionari preesistenti.
- ***Condizioni necessarie per l'attacco***: ogni sistema che dispone di un sistema di autenticazione è esposto a questo attacco.
- ***Probabilità di successo***: dipende dalla dimensione dello spazio delle credenziali.

- ***Aspetti facilitanti***: l'effort necessario per un attacco può essere sensibilmente ridotto da uno o più dei seguenti fattori:
  - password guessable: l'uso da parte degli utenti di password guessable aumenta la probabilità di successo degli attacchi basati su dizionario;
  - struttura delle credenziali: l'imposizione di una struttura semplice alle credenziali (ad esempio, password costituite da soli numeri o sole lettere, lunghezza non superiore a sei caratteri, ecc.) può diminuire sensibilmente la dimensione dello spazio di ricerca;
  - l'uso di messaggi di errore troppo informativi in caso di autenticazione fallita (ad esempio indicazioni che permettono di distinguere fra username errato e password errata) possono ridurre lo spazio di ricerca.
- ***Contromisure***[4]: gli attacchi di tipo brute force non possono essere prevenuti, ma esistono tecniche efficaci per ridurne drasticamente la probabilità di successo:
  - limitazione del numero massimo di tentativi di autenticazione falliti per ogni connessione;
  - adozione di controlli che vietano l'uso di password guessable o troppo semplici;
  - eliminazione dei messaggi di errori informativi.

### 6.3.2 Cross site scripting (XSS)

- ***Obiettivo***: furto d'identità ai danni di utenti di applicazioni web che fanno uso di cookie per la gestione delle sessioni.
- ***Attaccanti***: chiunque sia interessato al furto dell'identità di un utente autorizzato (che abbia stabilito una sessione con l'applicazione web).
- ***Descrizione***: si tratta di una tecnica che, mediante l'inserimento di elementi di scripting nei parametri inviati all'applicazione, provoca l'esecuzione degli stessi da parte del browser della vittima. In alcuni casi particolari le stesse tecniche e le stesse vulnerabilità applicative possono essere sfruttate per provocare l'esecuzione di codice sul server (esempio: Server Side Include, ecc.). Gli elementi di scripting causano l'invio dei cookie settati dall'applicazione target sul browser della vittima verso server un HTTP sotto il controllo dell'attaccante. Solitamente l'obiettivo dell'attacco è il cookie di sessione della vittima. La conoscenza di questo cookie permette infatti di sottoporre richieste all'applicazione utilizzando l'identità della vittima. Gli attacchi di cross site scripting sono possibili quando l'applicazione web restituisce al browser

[4] Le contromisure indicate in questo come in tutti gli altri casi sono da intendersi ovviamente come generiche. Caso per caso potranno essere o smentite, o confermate oppure rese più precise e puntuali.

(per normale logica di funzionamento o a causa di una condizione di errore) parametri sottoposti dall'utente in una precedente richiesta.

- ***Condizioni necessarie per l'attacco***: assenza di controlli sull'input ricevuto ed errori relativi all'escaping di metacaratteri nell'HTML ritornato al browser.
- ***Probabilità di successo***: questo attacco richiede l'uso di tecniche di social engineering per indurre la vittima a stabilire una sessione con l'applicazione target e sottoporre ad essa una richiesta contenente il codice malizioso. Frequentemente questo viene fatto per mezzo di email che invitano a seguire un link verso l'applicazione target. La probabilità di successo di tali attacchi è solitamente bassa.
- ***Aspetti facilitanti***: in alcuni casi è possibile inserire elementi di scripting in parametri che vengono salvati su database e restituiti all'utente ad ogni successiva richiesta (database XSS). Questo determina l'invio di cookie verso il server HTTP sotto il controllo dell'attaccante ogni volta che la vittima accede all'applicazione ed aumenta in modo considerevole la probabilità che l'attaccante riesca ad utilizzarlo con successo.
- ***Contromisure***: gli attacchi di tipo XSS possono essere neutralizzati mediante le seguenti tecniche:
  - filtraggio dei parametri in input, mediante filtri che eliminano dall'input i metacaratteri utilizzati in HTML e linguaggi di scripting client-side (<, >, apici, ecc.);
  - escaping dei metacaratteri contenuti nei parametri in input che devono essere inseriti in pagine HTML restituite al browser degli utenti.

### 6.3.3 SQL Injection

- ***Obiettivo***: gli attacchi basati su SQL injection possono avere diversi obiettivi:
  - accesso ad informazioni riservate memorizzate sui database server che costituiscono il data layer dell'architettura applicativa attaccata;
  - accesso non autorizzato all'applicazione, aggirando il meccanismo di autenticazione;
  - esecuzione di comandi sui server del data layer.
- ***Attaccanti***: utenti autorizzati che mirano ad accedere ad informazioni per le quali non possiedono diritti di accesso; utenti non autorizzati.
- ***Descrizione***: si tratta di tecniche di manipolazione dei parametri in input utilizzati dall'applicazione per eseguire query SQL sul database. Lo scopo è sovvertire la logica della query in modo da ottenere:
  - messaggi di errore contenenti informazioni sulla struttura del database utilizzato;
  - informazioni differenti da quelle che la query dovrebbe estrarre;

- recordset vuoti o tali da produrre un malfunzionamento dei meccanismi di autenticazione, allo scopo di accedere all'applicazione senza essere in possesso di credenziali valide;
- esecuzione di comandi di sistema tramite stored procedure.

- ***Condizioni necessarie per l'attacco***: mancanza di filtri di validazione dell'input, che eliminano dai parametri inviati dall'utente token pericolosi, come parole chiave riservate del linguaggio SQL (ad esempio, SELECT, OR, ecc.).
- ***Probabilità di successo***: fortemente dipendenti dalla logica applicativa.
- ***Aspetti facilitanti***: la visualizzazione, sul lato client, dei messaggi di errore relativi all'accesso al database permette all'attaccante di raccogliere informazioni sulla sua struttura, aumentando significativamente le probabilità di successo.
- ***Contromisure***: gli attacchi di tipo SQL injection possono essere neutralizzati mediante le seguenti tecniche:
  - filtraggio dei parametri in input, mediante filtri che eliminano dall'input token riservati e metacaratteri del linguaggio SQL;
  - gestione degli errori di accesso al layer di accesso ai dati, allo scopo di intercettare e bloccare la visualizzazione lato client dei messaggi di errore.

### 6.3.4 Path traversal

- ***Obiettivo***: browsing di directory presenti sul web server ma non appartenenti alle applicazioni web pubblicate su di esso, per le quali non è previsto l'accesso da parte degli utenti.
- ***Attaccanti***: questo tipo di attacco può essere portato da chiunque possa stabilire una connessione HTTP verso i server su cui è pubblicata l'applicazione.
- ***Descrizione***: un attacco di path traversal consiste nella sottomissione di richieste verso il web server per risorse il cui URL contiene path non appartenenti alle applicazioni web pubblicate su di esso. Poiché in generale tali path non sono noti all'attaccante, essi vengono specificati in forma relativa, partendo dalla posizione di risorse note ed utilizzando sintassi del tipo "../../" per navigare a ritroso il file system. Si noti che questo attacco non è in alcun modo correlato alla logica applicativa, ma sfrutta eventuali vulnerabilità del server HTTP.
- ***Condizioni necessarie***: queste tecniche possono essere utilizzate in presenza di web server sui quali non sono installate le security patch opportune; in ogni caso, la loro applicabilità non dipende dalla particolare applicazione pubblicata sul web server.
- ***Probabilità di successo***: dipende dall'accuratezza della manutenzione del web server.
- ***Aspetti facilitanti***: nessuno

- ***Contromisure***: aggiornamento dei web server mediante applicazione delle opportune patches.

### 6.3.5 OS command injection

- ***Obiettivo***: esecuzione di comandi di sistema sulle macchine su cui insiste l'applicazione.
- ***Attaccanti***: questo tipo di attacco può essere portato da chiunque possa stabilire una connessione HTTP verso i server su cui è pubblicata l'applicazione.
- ***Descrizione***: questo attacco può essere effettuato quando la logica applicativa utilizza dati forniti in input dall'utente come parametri per l'esecuzione di comandi di sistema. Se la logica applicativa non esegue correttamente il parsing di tali dati, è possibile provocare l'esecuzione di comandi aggiuntivi e/o differenti da quelli previsti dagli sviluppatori.
- ***Condizioni necessarie***: queste tecniche possono essere utilizzate in presenza di componenti dinamici che richiamano comandi di sistema senza effettuare un corretto parsing dei parametri di input.
- ***Probabilità di successo***: dipendenti dall'accuratezza della logica di validazione dei parametri in input.
- ***Aspetti facilitanti***: mancanza di funzionalità di filtraggio dell'output ritornato dopo l'esecuzione del comando di sistema.
- ***Contromisure***: gli attacchi di questo tipo possono essere neutralizzati eliminando dai parametri in input token riservati e metacaratteri potenzialmente pericolosi che potrebbero generare ambiguità per l'interprete dei comandi.

### 6.3.6 Cookie poisoning

- ***Obiettivo***: gli obiettivi possono essere molteplici; essi dipendono dalla logica dell'applicazione attaccata. In generale, le tecniche di cookie poisoning mirano a provocare comportamenti non previsti nell'applicazione attaccata in modo da poter interagire con essa in modi non previsti dal programmatore.
- ***Attaccanti***: gli attacchi di cookie poisoning possono provenire da qualsiasi utente sul cui browser l'applicazione setta cookie.
- ***Descrizione***: le tecniche di cookie poisoning consistono nella modifica dei dati contenuti nei cookie inviati dall'applicazione all'utente, allo scopo di produrre errori e/o portare l'applicazione in stati non correttamente gestiti quando i cookie sono restituiti al server. Per essere in grado di apportare le modifiche opportune, l'attaccante deve conoscere la logica con cui i dati contenuti nei cookie sono processati dall'applicazione.

- ***Probabilità di successo***: dipendenti dal livello di conoscenza da parte dell'attaccante della logica di elaborazione dei dati contenuti nei cookie.
- ***Aspetti facilitanti***: la memorizzazione nei cookie di parametri critici dal punto di vista della sicurezza è un errore comune, che rende pericolosi gli attacchi di *cookie poisoning*.
- ***Contromisure***: limitare l'uso dei *cookie* alla memorizzazione di informazioni non critiche; nel caso questo non sia possibile, devono essere utilizzate tecniche (ad esempio crittografia) per impedire la modifica dei *cookie*.

### 6.3.7 Forceful browsing

- ***Obiettivo***: accesso non autorizzato a pagine e/o funzionalità dell'applicazione.
- ***Attaccanti***: chiunque non sia in possesso di credenziali per accedere a tali pagine/funzionalità.
- ***Descrizione***: gli attacchi di *forceful browsing* consistono semplicemente nella sottomissione di richieste HTTP per URL corrispondenti a pagine protette, senza seguire il percorso di navigazione previsto dal programmatore e, in particolare, aggirando le pagine di autenticazione.
- ***Probabilità di successo***: dipendenti dal livello di conoscenza da parte dell'attaccante della struttura dell'applicativo. Tale livello può essere molto elevato per ex utenti che sono stati disabilitati.
- ***Aspetti facilitanti***: messaggi di errore non propriamente gestiti dall'applicazione possono contenere informazioni sulla struttura delle directory dell'applicazione sul web server e semplificare la costruzione degli URL da utilizzare per compiere l'attacco.
- ***Contromisure***: implementare una logica di controllo dello stato della sessione che impedisca l'accesso ad ogni parte dell'applicazione ad utenti non associati a sessioni autenticate.

### 6.3.8 Information leaking

- ***Obiettivo***: ottenere informazioni sul sistema da attaccare.
- ***Attaccanti***: chiunque possa navigare il sito.
- ***Descrizione***: vengono esaminati i sorgenti HTML delle pagine web ritornate dall'applicazione, allo scopo di individuare informazioni sensibili, come
  - password cablate nel codice;
  - commenti erroneamente lasciati dagli sviluppatori;
  - informazioni su versioni del software utilizzato e configurazione.
- ***Probabilità di successo***: dipendenti dal livello di *security-awareness* degli sviluppatori.
- ***Aspetti facilitanti***: nessuno.

- ***Contromisure***: eliminare dati sensibili dal codice HTML delle pagine web ritornate dall'applicazione.

# 7 Criticità di un assessment

## *7.1 Denial of service*

Il processo di eliminazione dei falsi positivi, dovuti ai *check* euristici effettuati dai sistemi di *vulnerability assessment* automatico, può richiedere il tentativo diretto di *exploiting* di un servizio, al fine di verificare l'effettiva presenza di una vulnerabilità, la possibilità di sfruttarla per prendere il controllo di un servizio, il suo impatto sulla sicurezza dei sistemi e dei dati in essi contenuti. Tuttavia, non tutte le classi di vulnerabilità richiedono l'utilizzo di *exploit* che possono compromettere la stabilità di un servizio o dell'intero sistema. Tipicamente, le vulnerabilità il cui tentativo di utilizzo può risultare in un D.o.S. sono quelle legate a problemi di *boundary check* e *integer overflow* (*stack/heap overflow*), *memory allocation*, *format string bug*, etc., il cui sfruttamento richiede la sovrascrittura di zone di memoria del processo contenenti strutture dati, indirizzi di ritorno, etc.

## *7.2 Perdita o inconsistenza di dati*

Alcune classi di attacco applicativo (es: *SQL Injection*, *Cross Site Scripting*, etc.) prevedono l'accesso non convenzionale o la manipolazione di dati persistenti, tipicamente immagazzinati in un database relazionale. In alcuni casi, l'eliminazione dei falsi positivi (vedi punto precedente), o addirittura la semplice rilevazione della vulnerabilità, richiede la modifica permanente dei dati persistenti. Ad esempio, per verificare la possibilità di cancellare una tabella da un database SQL, sfruttando dei permessi d'accesso poco restrittivi o una mancata validazione degli input, è richiesta l'effettiva cancellazione della tabella stessa. In altri casi, come ad esempio nelle vulnerabilità di tipo *Database Cross Site Scripting*, è richiesto l'inserimento di particolari *entries* malformate all'interno dei database utilizzati da un applicazione web-based. Questo potrebbe portare ad inconsistenze nel caso tali dati venissero utilizzati da un sistema di reportistica o *data-mining.*

## *7.3 File e processi zombie*

Durante una simulazione d'attacco completa, alcuni tipi di approccio richiedono l'upload sulla macchina target di particolari tools (*netcat, pwdump,* etc.) o l'esecuzione di particolari processi, per

permettere all'attaccante "simulato" di ottenere un pieno accesso alla macchina, per effettuare la cattura di dati sensibili o eliminare i log in maniera automatizzata, per elevare i propri privilegi, etc. In casi molto particolari non è possibile eliminare i file creati sulla macchina o i processi lanciati, senza un intervento diretto sul sistema da parte di un operatore.

# 8 Metodologia in caso di vincoli o divieti

Qualora i rischi connessi a particolari fasi dell'*assessment*[5] (eliminazione falsi positivi, simulazione d'attacco, etc.) non siano accettabili per il Cliente, è possibile ottenere i medesimi risultati utilizzando i seguenti tipi di approccio, unicamente a prezzo di una maggiore richiesta in termini di tempo:

- **Utilizzo sistemi di test:** E' possibile eliminare il rischio di potenziali disservizi causati dalle fasi di analisi più invasive (ad esempio i *Denial of Service* dovuti a tentativi di *exploiting* falliti), effettuando tali fasi sui sistemi di test. Questo tipo di attività deve essere preceduta da una verifica accurata dell'allineamento fra gli ambienti di test e di produzione. Nel caso non sia presente un ambiente di test, è possibile applicare delle procedure per ottenere una replica esatta dell'ambiente di produzione senza comprometterne l'operatività.
- **Verifica manuale:** Nell'eliminazione dei falsi positivi, in alternativa al tentativo diretto di *exploiting*, è possibile utilizzare un approccio di verifica manuale delle singole vulnerabilità rilevate dai prodotti di *assessment* automatico. Tale tipo di approccio prevede la verifica di presenza, e l'eventuale applicazione, di tutte gli aggiornamenti, *patch*, *best practice*, che possano eliminare o mitigare il problema riscontrato. Questa procedura, che deve essere comunque seguita per tutte le vulnerabilità che risultano effettivamente utilizzabili a scopi maliziosi, in questo caso deve essere applicata a tutte le criticità rilevate dai software di *scanning*.
- **Attacco manuale:** Negli attacchi di tipo applicativo è possibile eliminare o minimizzare il rischio connesso alla manipolazione dei dati persistenti non utilizzando software di analisi automatica. Le parti dell'applicazione che accedono a dati critici possono essere verificate manualmente eliminando il rischio di perdite accidentali, e permettendo l'immediato ripristino dei dati per cui è richiesta una manipolazione.

[5] Le fasi più rischiose ed invasive di un assessment sono svolte unicamente qualora il Cliente richieda una particolare accuratezza nei risultati e nella valutazione degli scenari d'attacco e degli impatti. I rischi connessi ad *vulnerability assessment* "generico" sono in genere talmente bassi da essere accettabili per qualsiasi sistema che non sia considerato particolarmente critico.

# 9 Tools utilizzati

Si elencano di seguito i tools che potrebbero essere utilizzati durante le attività. Si fa presente che il ruolo fondamentale in un'attività di assessment di qualsiasi tipo è dato dall'esperienza e dalla conoscenza di chi lo porta a termine; non è lo strumento che si utilizza che fa la differenza. Tant'è vero che abitualmente i tools automatici ricoprono solo una piccola parte (discovery e scanning) che è minimale rispetto al totale delle attività da intraprendere in un assessment professionale e di qualità.

## *9.1 Tools per l'assessment sistemistico*

Essendo le attività sistemistiche non richieste dal cliente, il presente paragrafo è stato inserito al solo scopo di fornire una conoscenza completa circa la strumentazione eventualmente utilizzata. La tipologia *system assessment* prevede più esperienza e conoscenza (e quindi attività manuali seguendo anche determinate guidelines) rispetto all'utilizzo di tools veri e propri.
Alcuni tra i tools utilizzati solitamente in questo ambito sono i seguenti:

- TripWire - System Integrity checker
- MBSA - Local vulnerability/patch checker
- RootkitRevealer - Sistema di rilevamento rootkit e trojan horse

## *9.2 Tools per l'assessment di rete e dei servizi*

Alcuni tra i tools utilizzati solitamente in questo ambito sono i seguenti:

- Internet Scanner - software vulnerability assessment generico
- Nessus - software vulnerability assessment generico
- Retina - software vulnerability assessment generico
- nmap - port/service scanner
- hydra - account brute force
- amap - service discovery/scanner
- ettercap - sniffing-MITM tool
- enum - windows resource enumeration
- Solarwinds - Network Discovery
- bluesnarf - Bluetooth analyzer/cracker
- kismet - Wireless LAN analyzer
- aircrack - Wireless LAN cracker
- john the ripper - password cracker

- l0pht crack 5 - password cracker
- ikescan - VPN discovery/assessment
- hping - network probing
- metasploit - exploit framework

### *9.3 Tools per l'assessment applicativo*

I principali tools in questo ambito sono gli scanner applicativi: si tratta di tools che eseguono in modo automatico la navigazione (crawling) delle pagine web dell'applicazione target, riducendo significativamente il tempo necessario a ricostruirne la struttura completa. Tali scanner eseguono inoltre una serie di test finalizzati ad evidenziare l'eventuale vulnerabilità dell'applicazione ad una serie di attacchi comuni.

In alternativa esistono anche

- Web proxy: sono tool di intercettazione del traffico fra browser e server applicativo, che permettono di analizzare e modificare header e body di ogni singola richiesta/risposta HTTP.
- Decompilatori ed analizzatori di codice: per un'analisi approfondita dei binari che compongono la parte client-side dell'applicazione, vengono utilizzati dei software in grado di risalire a porzioni del codice sorgente originale, ed altri strumenti in grado di rilevare tracce di programmazione insicura.

Alcuni tra i tools utilizzati solitamente in questo ambito sono i seguenti:

- Domino Scan II - software vulnerability assessment per Lotus Domino
- NgsSquirrel - software vulnerability assessment per database (Oracle, DB2, SQL Server)
- WebInspect - WEB application assessment tool
- AppScan - WEB application assessment tool
- Paros - WEB Proxy
- Nikto - WEB Server assessment tool
- OraScan - Oracle WEB Application auditing

# 10 Vincoli generali di progetto

Seguendo i vincoli espressamente richiesti dal cliente, si elencano di seguito i punti accettati da Hacking Team:

- l'attività sarà eseguita da personale dipendente Hacking Team
- per le attività di ethical hacking non saranno utilizzate tecniche di social engineering

- per le attività di penetration test non verranno utilizzate tecniche legate a infezione mediante virus
- le eventuali vulnerabilità di tipo DoS o DDoS verranno solo verificate ma non messe in atto
- i documenti consegnati (report per la direzione e report tecnico) verranno prodotti in lingua italiana (con l'esclusione di eventuali report prodotti da strumenti automatici rilasciati esclusivamente in lingua inglese)

# 11 Proposta e stima delle attività richieste

Per ogni attività richiesta verrà dedicato un paragrafo denominato attraverso gli stessi punti citati nella richiesta di offerta. In tale paragrafo si preciseranno:

- perimetro e vincoli: oggetto della relativa attività ed eventuali vincoli presenti
- descrizione delle attività: dettaglio delle operazioni svolte e dei passi metodologici
- stima dell'effort: tabella contenente i dati di stima richiesti
  - stima delle attività: giorni/uomo stimati per il completamento della relativa attività[6]
  - stima dell'elapsed: tempo solare garantito[7] per l'effettuazione ed il completamento delle attività, comprensivo della stesura della documentazione. A seconda della disponibilità di risorse e della disponibilità del cliente potrebbe essere in alcuni casi compresso
  - tempi di attivazione[8]: tempo previsto per l'approvvigionamento e per la partenza dei lavori a partire dall'eventuale ordine del cliente
  - numero e tipologia delle figure professionali coinvolte
- piano di massima: gantt sintetico rappresentante le macro-attività necessarie al completamento della relativa attività

[6] Si intende la totalità delle giornate impiegate per completare l'attività a prescindere dal numero di figure professionali impiegate

[7] A patto che gli eventuali slittamenti temporali non dipendano da inefficienze del servizio Hacking Team o da cause di forza maggiore

[8] Si intende il tempo massimo di attesa; quindi, a seconda della disponibilità delle figure professionali richieste, l'attività potrebbe iniziare **entro** la stima indicata

## *11.1 Punto (a)*

### 11.1.1 Perimetro e vincoli

L'oggetto dell'attività è indicato essere un parco macchine di massimo 15 server e 10 apparati di rete. Gli attacchi portati durante tali attività non dovranno essere condotti attraverso l'infezione virale o attraverso l'utilizzo di tecniche di social engineering. Inoltre saranno evitati azioni di Dos e DDos, adottando l'accortezza di menzionare soltanto la possibile presenza di un simile minaccia.
Nella richiesta pervenuta dal cliente, fa parte anche l'analisi del traffico di rete nell'ottica di individuare eventuali trasmissioni di dati non autorizzati o legati agli accessi[9].

### 11.1.2 Descrizione delle attività

L'attività richiesta e riguardante esclusivamente il perimetro descritto nel paragrafo precedente, consiste in un penetration test / vulnerability assessment dall'esterno[10]. La tecnica ed i passi utilizzati per il suo completamento sono descritti nel paragrafo 6.2.
Per soddisfare appieno ai requisiti richiesti in merito alla verifica di informazioni trasmesse in maniera non autorizzata, sarebbe opportuno[11]:

- inserire uno sniffer per qualche giorno ed analizzare i risultati nell'ottica di verificare se esistono dei trasferimenti di dati non consentiti (in questo caso si incorre in un possibile falso negativo: non rilevare pacchetti di informazioni transitanti non è una condizione sufficiente a poter garantire che le macchine del perimetro non siano in qualche modo prive di minacce; potrebbe invece dire che in quei due giorni non si sono verificati trasferimenti illeciti[12]

*oppure*

- eseguire un assessment sistemistico (consultare il paragrafo 6.1 per la tecnica di analisi) sulle macchine perimetrali individuate nel perimetro del paragrafo precedente

### 11.1.3 Stima dell'effort

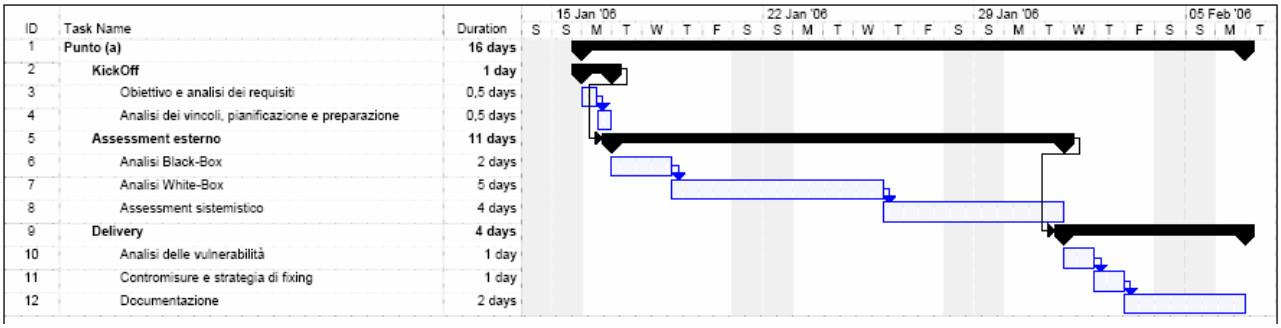
Le stime della presente attività possono essere sintetizzate nella seguente tabella.

---

[9] Questa attività non può essere coperta da un assessment dall'esterno, il quale si renderebbe conto di queste vulnerabilità solo una volta preso possesso di una macchina. Quanto espressamente richiesto dal cliente può essere effettivamente eseguito da un **assessment sistemistico** sulle macchine oppure mediante uno **sniffing di rete** vero e proprio sul link verso internet (o genericamente verso l'esterno) che rilevi una tale anomalia.
[10] La stesura della documentazione di progetto è compresa nella stima indicata
[11] Le attività citate sono espresse nella stima della tabella successiva (nello stesso paragrafo) come opzionali e quindi rimarranno opzionali anche nell'offerta economica relativa allo stesso punto (a)
[12] E' per tale motivazione che si raccomanda la seconda strategia di intervento (assessment sistemistico), più dispendiosa ma più sicura e completa

| Parametro | Stima |
|---|---|
| Stima delle attività | Black-box: 2 giornate/uomo<br>White-Box: 5 giornate uomo<br>Attività **opzionale** di sniffing: 2 giornate/uomo<br>Attività **opzionale** di assessment sistemistico: 4 giornate/uomo (scelta **al posto** della attività precedente)<br>Attività a complemento (vedere il gantt successivo): 5 giornate/uomo |
| | **Totale: 12 giornate/uomo + 4 (opzionali)** |
| Stima dell'effort (in elapsed) | Black-box + White-box: 4 settimane solari<br>Con l'aggiunta dell'attività opzionale: 5 settimane solari |
| Tempi di attivazione | 2 settimane solari |
| Figure professionali coinvolte | 1 Network Security Analyst<br>1 System Security Analyst (nel caso di attività opzionale di assessment sistemistico) |

### 11.1.4 Piano di massima




## 11.2 Punto (b) e Punto (c)

### 11.2.1 Perimetro e vincoli

L'oggetto del presente assessment di sicurezza consiste in massimo 30 apparati interni della rete Toro Assicurazioni e dei relativi servizi erogati dalle macchine stesse. Si tratta di server aventi le seguenti funzioni:

- Remote Access Server
- Accesso via GPRS
- Eventuali Access Point wireless
- Il sistema di posta elettronica
- I sistemi di accesso e profilazione degli utenti

Si considerano i medesimi vincoli dell'assessment esterno (consultare il paragrafo 11.1.1).

### 11.2.2 Descrizione delle attività

L'attività richiesta e riguardante esclusivamente il perimetro descritto nel paragrafo precedente, consiste in un penetration test / vulnerability assessment dall'interno[13]. La tecnica ed i passi utilizzati per il suo completamento sono descritti nel paragrafo 6.2.

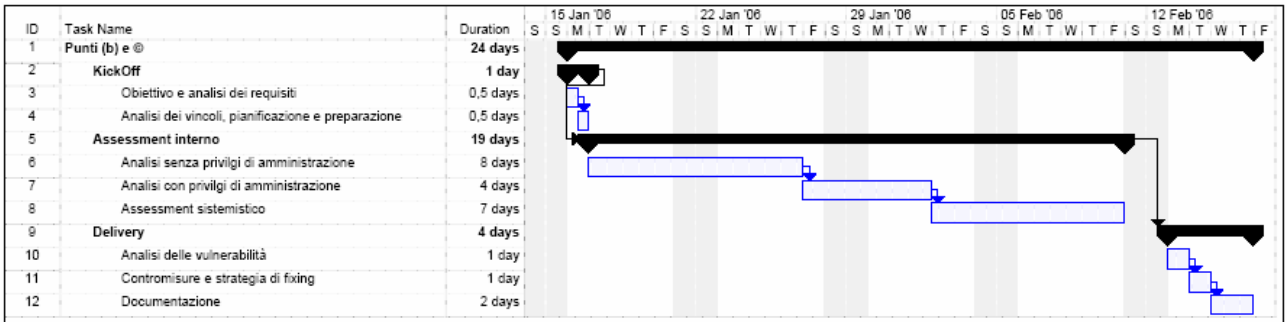
Non sono presenti richieste di analisi del traffico o di verifica del trasferimento illecito di informazioni; ciononostante si consiglia un'attività opzionale di assessment sistemistico sulle macchine (le cui attività sono indicate nel paragrafo 6.1) indicate, per completare il controllo di sicurezza che in alternativa risulterebbe privo della componente locale.

### 11.2.3 Stima dell'effort

Le stime della presente attività possono essere sintetizzate nella seguente tabella.

| ***Parametro*** | ***Stima*** |
|---|---|
| Stima delle attività | Assessment interno: 12 giornate/uomo<br>Attività **opzionale** di assessment sistemistico: 7 giornate/uomo<br>Attività a complemento (vedere il gantt successivo): 5 giornate/uomo |
| | **Totale: 17 giornate/uomo + 7 (opzionali)** |
| Stima dell'effort (in elapsed) | Assessment interno: 5 settimane solari<br>Con l'aggiunta dell'attività opzionale: 7 settimane solari |
| Tempi di attivazione | 2 settimane solari |
| Figure professionali coinvolte | 1 Network Security Analyst<br>1 System Security Analyst (nel caso di attività opzionale di assessment sistemistico) |

### 11.2.4 Piano di massima




[13] La stesura della documentazione di progetto è compresa nella stima indicata

## 11.3 Punto (d)

### 11.3.1 Perimetro e vincoli

Si richiede di esaminare la connessione VPN che collega le agenzie alla sede centrale. Non si tratta di valutare la sicurezza del sistema VPN implementato ma di verificare se attraverso di esso si rendano visibili vulnerabilità utilizzabili da personale attestato sulla rete agenziale.

### 11.3.2 Descrizione delle attività

L'attività richiesta è quindi quella di eseguire un assessment di sicurezza di reti e servizi partendo da una postazione di test (allestita preventivamente dal cliente) che simula in tutto e per tutto un collegamento di agenzia. Le attività sono quindi quelle indicate dal paragrafo 6.2.
Le attività saranno condotte simulando:

- la presenza nella rete di agenzia (solitamente non in gestione del cliente) di personale estraneo o di personale virtuale presente in rete a seguito di un attacco attraverso un collegamento internet o modem di agenzia: approccio black-box, senza credenziali
- la presenza di un utente d'agenzia malizioso: approccio white-box, con credenziali

Come consulenti informatici e conoscitori di ambienti informatici assicurativi, poniamo in evidenza due punti a nostro giudizio fondamentali:

1. non si richiede l'analisi di sicurezza di server o client (assessment sistemistico) eventualmente configurati ed assegnati alle agenzie agenti da parte del cliente ed amministrati da remoto; si suppone quindi che l'agenzia acquisti e gestisca l'intero parco macchine in completa autonomia e che non ci sia la presenza di un server locale in comunicazione con i sistemi centrali
2. non si richiede un'analisi applicativa ma solo un assessment di rete e servizi

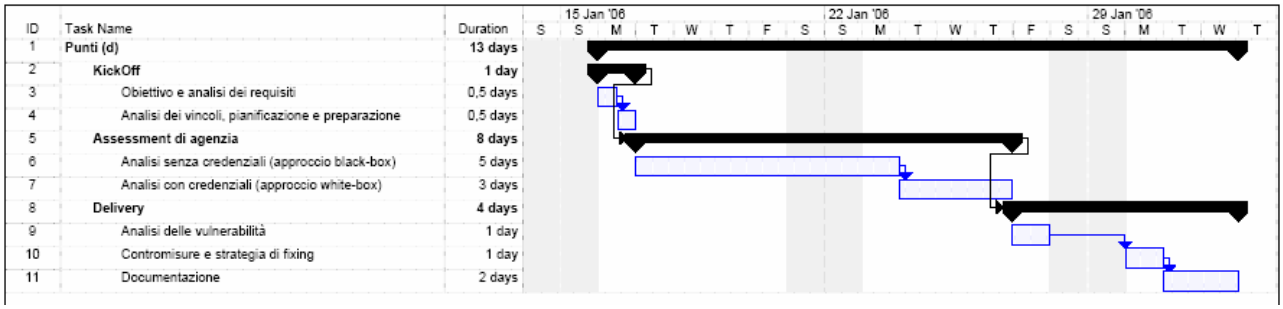
Molto spesso capita che le principali vulnerabilità e minacce non si riscontrino nell'architettura perimetrale a difesa della sede centrale ma siano presenti nei client e server dati agli agenti oppure nei bachi di natura applicativa risiedenti nelle applicazioni (tipicamente web-based) utilizzate all'interno dell'agenzia e comunicanti con la sede centrale.
La mancanza di informazioni a riguardo, ci impedisce di inserire tali attività anche come opzionali nel computo totale della stima.

### 11.3.3 Stima dell'effort

Le stime della presente attività possono essere sintetizzate nella seguente tabella.

| *Parametro* | *Stima* |
|---|---|
| Stima delle attività | Assessment d'agenzia senza credenziali: 5 giornate/uomo<br>Assessment d'agenzia con credenziali: 3 giornate/uomo<br>Attività a complemento (vedere il gantt successivo): 5 giornate/uomo |
| | **Totale: 13 giornate/uomo** |
| Stima dell'effort (in elapsed) | 4 settimane solari |
| Tempi di attivazione | 3 settimane solari |
| Figure professionali coinvolte | 1 Network Security Analyst<br>1 Network Analyst |

### 11.3.4 Piano di massima




## *11.4 Punto (e) NON IN OFFERTA*

### 11.4.1 Perimetro e vincoli

Le applicazioni da testare sono 5: OnLine, Coweb, S.INTE.SI, ToroTarga (tutte web-based) e SAP.
In mancanza di informazioni più precise, si ipotizza (su consiglio del cliente) di dover analizzare:

- 5 form di login, una per ognuna delle 5 applicazioni da testare
- 100 form di varia natura

Considerando che, oltre all'assessment applicativo, si richiede un'analisi del codice sorgente disponibile[14], le attività verranno stimate sulla base di un numero massimo di 50.000 linee di codice complessivo (comprendente quindi tutte e quattro le applicazioni web-based).
Si ritiene opportuno precisare che l'eventuale aumento di form e linee di codice rispetto a quelle non comporta un aumento **lineare** della stima di giornate necessaria.

[14] Questo ovviamente esclude dall'analisi del codice l'applicazione SAP che è un prodotto commerciale, proprietario e quindi senza la possibilità di avere il codice sorgente.

### 11.4.2 Descrizione delle attività

L'attività richiesta e riguardante esclusivamente il perimetro descritto nel paragrafo precedente, consiste in un penetration test / vulnerability assessment applicativo dall'interno[15]. La tecnica ed i passi utilizzati per il suo completamento sono descritti nel paragrafo 6.3.

Le attività previste per l'assessment applicativo (5 form di login e 100 form) sono dettagliate nella seguente tabella.

| *Test* | *Modalità di esecuzione* |
|---|---|
| **Gestione login** | I test sul form di login sono svolti manualmente. Comprendono test di resilienza al brute forcing, alla SQL injection (per bypassare il controllo delle credenziali) e la verifica della corretta gestione dei messaggi di errore |
| **Assegnazione diritti utente** | Questi test vengono svolti manualmente, controllando la corretta gestione (mediante variabili di sessione) dei parametri di autorizzazione associati ad ogni richiesta HTTP |
| **Gestione password** | Questi test vengono svolti manualmente, controllando il traffico HTTP fra browser e server |
| **Gestione errori** | Questi test vengono svolti con l'ausilio di tool automatici che eseguono operazioni di fuzzing sui paramentri inviati al server dai form dell'applicazione |
| **Gestione sessioni** | Questi test vengono svolti manualmente, controllando il funzionamento del meccanismo di tracking delle sessioni |
| **Gestione cookie** | Questi test vengono svolti con l'ausilio di tool automatici per la verifica della non-predicibilità dei cookie. Si controlla inoltre, manulamente, che i cookie memorizzati sul browser dell'utente non contengano informazioni sensibili e/o manipolabili |
| **Gestione cache browser** | Questi test sono svolti manualmente e comprendono la verifica della corretta terminazione delle sessioni per impedire accessi indesiderati mediante le funzioni "Back" o "Refresh" del browser |
| **Attacchi al DB** | Questi test vengono svolti con l'ausilio di tool automatici per l'identificazione di parametri sfruttabili per attacchi di tipo SQL Injection |

[15] La stesura della documentazione di progetto è compresa nella stima indicata

Le attività previste per l'analisi del codice sorgente (50.000 linee) sono dettagliate nella seguente tabella[16].

| *Test* | *Modalità di esecuzione* |
|---|---|
| Informazioni codificate all'interno di file HTML | Questi test includono l'uso di strumenti automatici di parsing e l'analisi manuale di porzioni di codice segnalate come sospette |
| Validazione dell'input | Questi test vengono effettuati con l'ausilio di tool automatici per il tracciamento del percorso dei parametri di ingresso all'interno dei moduli applicativi |
| Aggiramento SSL | Questi test sono svolti manualmente |
| Query dinamiche non precompilate e SQL injection | Questi test sono svolti con l'ausilio di tool automatici per l'identificazione di pattern noti corrispondenti all'uso di funzioni per l'interrogazione dei DB |

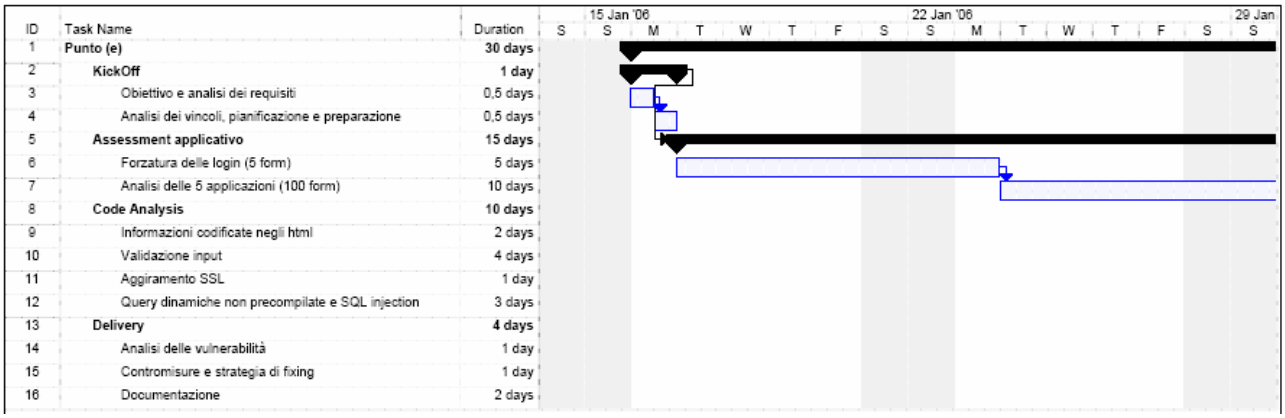
### 11.4.3 Stima dell'effort

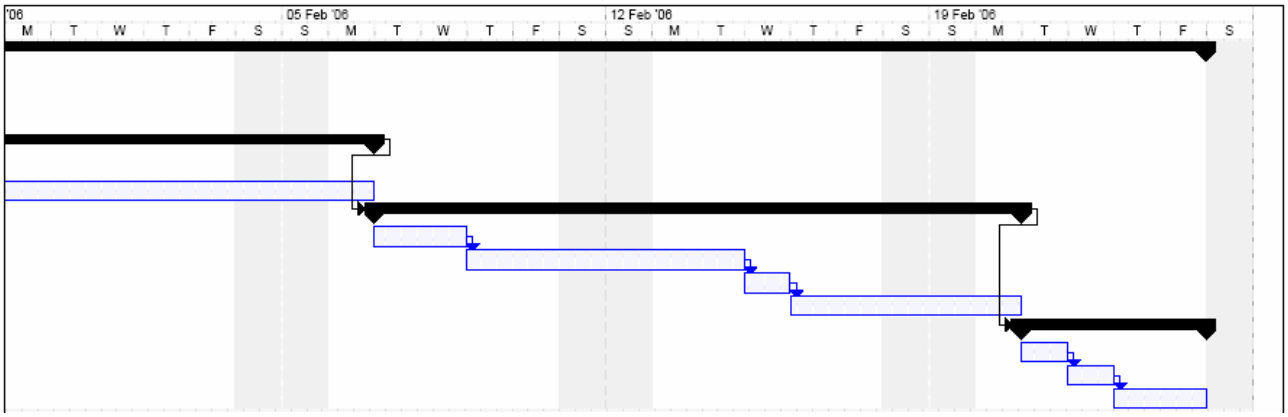
| *Parametro* | *Stima* |
|---|---|
| Stima delle attività | Assessment applicativo delle 5 applicazioni indicate nel perimetro: 15 giornate/uomo<br>Code Analysis del codice sorgente disponibile (50.000 linee java): 10 giornate/uomo<br>Attività a complemento (vedere il gantt successivo): 5 giornate/uomo |
| | **Totale: 30 giornate/uomo** |
| Stima dell'effort (in elapsed) | 9 settimane solari |
| Tempi di attivazione | 3 settimane solari |
| Figure professionali coinvolte | 1 Application Security Analyst<br>1 Application Analyst |

[16] Le attività di analisi del codice sorgente previste prendono in considerazione esclusivamente le esigenze scritte dal cliente nel documento di richiesta. Ciò non toglie che se dovessero emergere altri tipi di vulnerabilità, queste saranno opportunamente inserite nella documentazione finale e coperte dalla strategia di intervento.

### 11.4.4 Piano di massima

| ID | Task Name | Duration |
|---|---|---|
| 1 | Punto (e) | 30 days |
| 2 | KickOff | 1 day |
| 3 | Obiettivo e analisi dei requisiti | 0,5 days |
| 4 | Analisi dei vincoli, pianificazione e preparazione | 0,5 days |
| 5 | Assessment applicativo | 15 days |
| 6 | Forzatura delle login (5 form) | 5 days |
| 7 | Analisi delle 5 applicazioni (100 form) | 10 days |
| 8 | Code Analysis | 10 days |
| 9 | Informazioni codificate negli html | 2 days |
| 10 | Validazione input | 4 days |
| 11 | Aggiramento SSL | 1 day |
| 12 | Query dinamiche non precompilate e SQL injection | 3 days |
| 13 | Delivery | 4 days |
| 14 | Analisi delle vulnerabilità | 1 day |
| 15 | Contromisure e strategia di fixing | 1 day |
| 16 | Documentazione | 2 days |




## *11.5 Punto (f) NON IN OFFERTA*

### 11.5.1 Perimetro e vincoli

Il cliente non è in possesso di dispositivi wireless, tuttavia richiede un'attività di discovery con l'obiettivo di identificare eventuali sistemi non autorizzati. La stima effettuata è relativa ad un building (discovery ed analisi della sicurezza wireless); in presenza di più building, la stima e quindi la quotazione economica subirà un incremento lineare.

### 11.5.2 Descrizione delle attività

Un attacco basato sul discovery di access point avviene utilizzando degli analizzatori di traffico x802.11b. Infatti, ogni qualvolta un access point è presente, emette segnali ai potenziali client.
Una volta rilevato il traffico, si analizza la tipologia di cifratura utilizzata: wep o altro tipo. Nel caso ci sia cifratura, vi sono degli attacchi crittanalitici atti a scoprire la password a protezione della rete. Una volta che il client ha trovato un access point disponibile, cerca di scambiare la password per

poter comunicare con lui; una volta associati all'access point si potrebbe quindi ottenere un accesso a segmenti di rete e risorse differente da quello definito ed implementato.

In sintesi un'attività di wardriving consiste nel seguire le aree perimetrali di un edificio target alla ricerca di segnali wireless, quindi la stima dell'effort è lineare al numero di building da perlustrare. Inoltre, una volta trovato un apparato in ricezione o trasmissione, scatterà la fase di analisi descritta di seguito.

Non essendo ufficialmente presente alcun dispositivo wireless, non si eseguiranno le principali tecniche per verificare la presenza di vulnerabilità legate a tale protocollo. Qualora ce ne fosse bisogno (non è però stato quotato nella presente stima), le attività da eseguire sarebbero le seguenti:

- Attacchi di inserimento: consistono nella distribuzione incontrollata e non autorizzata di periferiche wireless e/o sulla creazione di reti wireless abusive, aggirando qualsiasi tipo di revisione architetturale.
- Intercettazione e monitoraggio non autorizzato del traffico: come nelle reti a cavo, è possibile intercettare e monitorare (*sniffare*) il traffico sulle reti 802.11[x]. Il punto di forza di questo attacco rispetto a un ambiente *wired* è che l'attaccante non ha bisogno di compromettere un sistema collegato alla rete per depositare un agente o un Trojan che faccia da *sniffer*. Tutto quello di cui si ha bisogno è riuscire a raggiungere la portante dei segnali usati dai sistemi *Wifi*. Visto che il segnale viene distribuito in maniera circolare sui tre assi dimensionali, il risultato è che questo può essere intercettato da posizioni esterne all'azienda o da un piano all'altro del palazzo.
- Jamming: gli attacchi di tipo *Denial of Service* possono essere facilmente applicati all'ambiente wireless: in questo caso la connettività è compromessa con l'iniezione di traffico illegittimo o disturbo della frequenza di trasmissione. Un eventuale attaccante con un buon equipaggiamento, potrebbe '*inondare*' (*flooddare*) il segnale, corrompendo lo *stream* dati fino alla caduta del servizio.
- Attacchi da client a client: gli standard prevedono che due client *wireless* possano colloquiare direttamente tra loro, senza utilizzare l'access point del loro *Service Set*. Di conseguenza gli utenti, hanno bisogno di essere protetti non solo dai rischi esterni, ma anche da elementi sconosciuti.
- Attacchi brute force all'access point: Diversi sistemi di distribuzione usano una singola chiave o password per autenticare tutti i *client*. Il *brute forcing,* tramite dizionario o tentativi

sequenziali, consente l'accesso al dispositivo di accesso ed di ottenere comodamente tutti i dati utente.

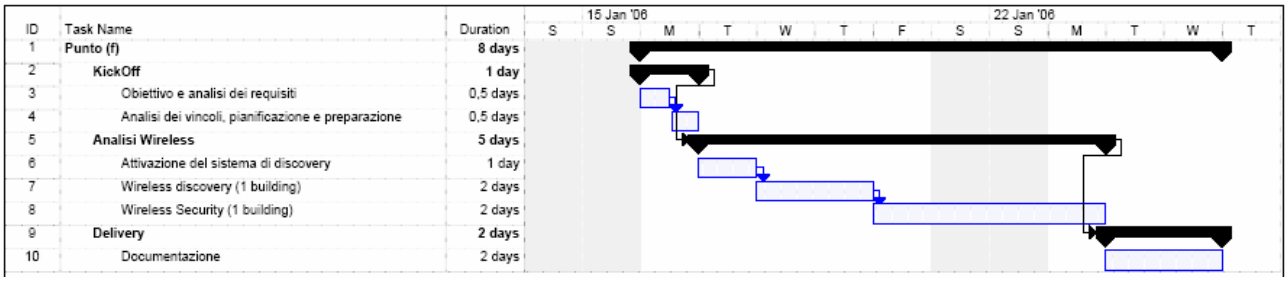
- Attacchi crittografici: Lo standard 802.11b usa un sistema di autenticazione chiamato *WEP*. Questo standard è potenzialmente soggetto a diversi tipi di attacco:
  - passivo, basato su analisi statistica del traffico
  - attivo, con iniezione di nuovo traffico da una stazione non autorizzata, basato sull'analisi del testo in chiaro passante
  - attivo, basato sulla compromissione dell'access point
  - attivo, tramite il monitoraggio continuato del traffico in un certo lasso temporale dell'ordine di qualche giorno, permettendo la decifrazione in tempo reale di tutto il traffico
- Errata configurazione: di solito i sistemi di accesso vengono distribuiti con una configurazione standard per una facile messa in produzione ed un utilizzo immediato. Questa tendenza porta spesso a definire configurazioni di default in cui la sicurezza viene posta in secondo piano.

Capire come funzionano gli attacchi e utilizzare queste informazioni per prevenirli, sono passi fondamentali nella stesura di una policy di sicurezza per una qualsiasi soluzione wireless.

### 11.5.3 Stima dell'effort

| ***Parametro*** | ***Stima*** |
|---|---|
| Stima delle attività | Wireless Discovery: 2 giornate/uomo per building<br>Wireless Security: 2 giornate/uomo per building<br>Attività a complemento (vedere il gantt successivo): 4 giornate/uomo |
| | **Totale: 4 giornate/uomo per building + 4 di complemento (come da gantt)** |
| Stima dell'effort (in elapsed) | 2 settimane solari |
| Tempi di attivazione | 2 settimane solari |
| Figure professionali coinvolte | 1 Network Security Analyst |

### 11.5.4 Piano di massima



| ID | Task Name | Duration |
|---|---|---|
| 1 | Punto (f) | 8 days |
| 2 | KickOff | 1 day |
| 3 | Obiettivo e analisi dei requisiti | 0,5 days |
| 4 | Analisi dei vincoli, pianificazione e preparazione | 0,5 days |
| 5 | Analisi Wireless | 5 days |
| 6 | Attivazione del sistema di discovery | 1 day |
| 7 | Wireless discovery (1 building) | 2 days |
| 8 | Wireless Security (1 building) | 2 days |
| 9 | Delivery | 2 days |
| 10 | Documentazione | 2 days |



## *11.6 Punto (g)*

### 11.6.1 Perimetro e vincoli

Linee guida e “best practices”per lo sviluppo di applicazioni sicure su protocollo http per ambienti J2EE.

### 11.6.2 Descrizione delle attività

L’attività comprende la preparazione di un documento che descrive linee guida e *best practices* per lo sviluppo di applicazioni web “sicure”.

Il documento sarà organizzato con:

- Introduzione al problema della sicurezza applicativa evidenziandone le peculiarità che lo distinguono dal più tradizionale settore della sicurezza di rete e dei sistemi operativi.
- Presentazione di una classificazione degli errori di progettazione e sviluppo delle applicazioni web su cui si basano le linee guida e le best practices proposte.
- Criteri di derivazione di linee guida e best practices e descrizione delle basi metodologiche ed i principi fondamentali di sicurezza a cui si fa riferimento. I paragrafi 5 e 6
- Descrizione di come, tramite l’applicazione al caso concreto delle applicazioni web di tali criteri, si giunga alla definizione di una metodologia “security-aware” per le fasi di progetto e sviluppo.

### 11.6.3 Stima dell’effort

| ***Parametro*** | ***Stima*** |
|---|---|
| Stima delle attività | Preparazione documento linee guida e presentazione: 4 giornate/uomo |
| | **Totale: 4 giornate/uomo** |

| | |
|---|---|
| Stima dell'effort (in elapsed) | 2 settimane solari |
| Tempi di attivazione | 2 settimane solari |
| Figure professionali coinvolte | 1 Senior Security Analyst |

## *11.7 Punto (h) NON IN OFFERTA*

### 11.7.1 Perimetro e vincoli

La richiesta di tale punto prevede la creazione di una struttura per il monitoraggio degli eventi potenzialmente dannosi, per la prevenzione e la gestione degli incidenti di sicurezza all'interno del Gruppo Toro Assicurazioni. In sostanza, si richiede la definizione di un sistema di incident handling composto di organizzazione e procedure[17].

Si propone quindi uno studio della realtà aziendale e dei suoi obiettivi da completare congiuntamente con lo stesso personale interno in modo da perseguire il seguente scopo[18]:

- impostare l'organigramma del team di incident response che dovrà gestire il sistema di gestione degli incidenti, definendo
  - ruoli e responsabilità e quindi chi fa cosa e chi decide cosa
  - flussi di comunicazione interni
  - flussi di comunicazione esterni (in caso di servizio completamente o parzialmente in outsourcing)
- capire le risorse che l'azienda vuole proteggere
  - analisi delle risorse critiche
  - analisi dei dati sensibili
- definire i meccanismi e le procedure di gestione del sistema (documenti organizzati a schede veloci da leggere e semplici da capire)
  - procedure di prevenzione degli incidenti
  - procedure di gestione degli incidenti
  - procedure di gestione degli incidenti nuovi e delle eccezioni
  - procedure di risposta agli incidenti
- definire i meccanismi e le procedure di aggiornamento del sistema
  - procedure di aggiornamento delle politiche aziendali

[17] Non è richiesta, e quindi neanche stimata, l'implementazione tecnologica di un sistema di incident handling. Lo studio è esclusivamente riferito alla progettazione organizzativa e procedurale.

[18] Il perimetro effettivo delle attività e la definizione dello scopo potranno essere ritagliati nella fase iniziale dell'attività a seconda delle esigenze e del volere del cliente

  - procedure di upgrade delle macchine
  - procedure di aggiornamento del database delle signature
- definire i meccanismi e le procedure di controllo del sistema
  - procedure di monitoraggio del sistema
  - procedure di tuning (affinamento)
- definire i meccanismi e le procedure straordinarie
  - cambiamento della rete o delle piattaforme, traslochi, nuovi indirizzamenti, nuovo personale, cambiamento organigramma
  - procedure di backup delle risorse critiche e dei dati sensibili ed eventualmente (se non presente) definizione di procedure di recovery in caso di attacco
- impostare il sistema documentale
  - tipo della documentazione (digitale, cartaceo...)
  - locazione della documentazione
  - reperibilità della documentazione (raccoglitori, schede, intranet...)

### 11.7.2 Descrizione delle attività

Le attività che caratterizzano il processo di risposta a un incidente, e che dovranno essere prese in considerazione, analizzate e definite, possono essere riassunte nei seguenti punti:

- Prevenzione: ogni programma di incident response dovrebbe cercare di prevenire eventuali incidenti utilizzando strumenti adeguati allo scopo, tra cui firewall, antivirus, IDS e congiuntamente avendo delle procedure di prevenzione ed aggiornamento
- Pianificazione: è necessario identificare i ruoli e le responsabilità delle persone, documentando le policy aziendali e le procedure da adottare in caso di violazioni
- Rilevazione attiva e/o passiva: è importante capire quando si è verificata una violazione. Solo del personale tecnico ben preparato, col supporto di strumenti specifici e di procedure dettagliate, può distinguere tra un rischio reale ed un falso positivo
- Analisi: quando una violazione ha avuto luogo è necessario raccogliere e analizzare tutte le informazioni a disposizione per comprendere il problema in tutti i suoi aspetti
- Contenimento: prima ancora di ripristinare uno stato sicuro è necessario isolare il problema e comprendere come porvi rimedio
- Ripristino: una politica di backup unita a procedure specifiche di restore devono essere studiate a priori per consentire il ripristino delle normali attività aziendali

- Postmortem: riesaminare l'accaduto, rivedere eventualmente tutte le fasi precedenti e se necessario modificare o aggiungere procedure/meccanismi per evitare la ripetibilità dell'evento

Per ognuno degli aspetti elencati, occorrerà definire un sistema documentale di **organizzazione** e **procedure** (che alla fine è quanto richiesto dal cliente) che descriverà e regolamenterà il sistema di gestione degli incidenti del cliente.
Una corretta pianificazione ha tra i suoi obiettivi principali:

- definire le risorse aziendali necessarie per realizzare un sistema di incident handling: umane, economiche e tecnologiche
- definire le policy aziendali , i ruoli e le competenze
- definire le procedure aziendali da seguire

La documentazione rappresenta un punto vitale di ogni processo aziendale, e in quanto tale, un sistema di incident handling, richiede uno sforzo perché essa risulti precisa, dettagliata e allo stesso tempo comprensibile. Sarà perciò necessario stabilire dove e come queste specifiche informazioni verranno memorizzate: in forma cartacea, piuttosto che elettronica. Anche l'organizzazione della documentazione risulta importante perché questa risulti rapidamente reperibile: si pensi al vantaggio di una gestione indicizzata delle informazioni. In questo modo in caso di effettive violazioni si minimizzerà sia il rischio di commettere errori, sia il ritardo dovuto alla ricerca delle informazioni desiderate.

Una volta operata la scelta del personale che costituirà il response team, è importante definire i ruoli e le competenze delle figure coinvolte nel processo di incident response. Definire ruoli e responsabilità è tanto più importante, tanto maggiore è la complessità della struttura di un'azienda. La definizione di un organigramma deve essere documentata e resa disponibile, in modo da identificare con chiarezza le persone coinvolte.
La conoscenza approfondita sia delle policy, sia del sistema informatico aziendale risultano importanti: ancora una volta emerge la necessità di documentare le informazioni relative alle policy e agli assets aziendali. Per queste fasi non esistono, o comunque sono limitate le procedure predefinite, infatti l'insieme delle violazioni o intrusioni possibili è talmente vasto che non è pensabile definire una procedura universalmente valida per distinguere tra una violazione/intrusione vera o falsa.
Le procedure che invece devono essere ben documentate riguardano il processo di notifica:

- Segnalazione dell'accaduto: è importante che sia chiaro chi deve essere informato (Manager, Public Relation Team) e di conseguenza la definizione di una hotline/presidio risulta in molti casi un must
- Tempi e ordine: bisogna definire il processo di segnalazione in termini di urgenza e cronologia
- Contenuto delle informazioni: quali informazioni in merito alla situazione ogni figura coinvolta deve conoscere e quali invece no
- Canali di comunicazione: informazioni riservate useranno canali sicuri, magari crittografati, informazioni urgenti canali prioritari, cercapersone, ...

Il fattore tempo può rivelarsi decisivo in queste fasi, e di conseguenza tali procedure devono essere facilmente reperibili. Inoltre, quest'ultime dovrebbero essere anche verificate di tanto in tanto al fine di rendere tali processi il più possibile automatici. Per fare ciò può risultare utile definire dei Fire Drill, ossia simulazioni di violazioni, che hanno lo scopo di misurare l'efficienza del incident response team e delle procedure definite.
Non per ultimo, occorrerà definire le procedure di contenimento, ripristino, maintenance e update.

### 11.7.3 Stima dell'effort

Considerando la mancanza di informazioni complete e la complessità della stima di un'attività prettamente consulenziale di affiancamento, si ritiene opportuno procedere commercialmente mediante l'utilizzo di giornate/uomo.

| ***Parametro*** | ***Stima*** |
|---|---|
| Stima delle attività | Non stimabile senza ulteriori informazioni |
| Stima dell'effort (in elapsed) | Non essendoci una stima totale delle giornate, questo parametro assume significato nullo |
| Tempi di attivazione | 3 settimane solari |
| Figure professionali coinvolte | 1 Security Manager |

### 11.7.4 Piano di massima

Essendo l'attività quotata a giornate, non è possibile prevedere a priori un piano di intervento. Sarà concordato insieme al cliente al momento della partenza delle attività, quando si avranno maggiori informazioni sugli obiettivi e sull'infrastruttura organizzativa, procedurale e tecnologica su cui basare la definizione di un piano di incident handling corredato da procedure.

### *11.8 Piano di lavoro complessivo*

Considerando che tutte le attività richieste dal cliente e descritte nel capitolo 11 sono autocontenute, il piano di lavoro complessivo è dato dall'unione dei gantt descritti nei paragrafi 11.1.4, 11.2.4, 11.3.4, 11.4.4, 11.5.4, **Errore. L'origine riferimento non è stata trovata.** e 11.7.4. Eventuali restringimenti temporali sull'elapsed potranno essere confermati in sede di inizio lavori, in relazione al carico di lavoro esistente.

# 12 Template o report di esempio

Si allegano al presente documento due report opportunamente anonimizzati concernenti le due tipologie principali di attività: ethical hacking (titolo: Ethical Hacking) e application assessment (titolo: Analisi applicativa):

- Appendice 1 - Ethical Hacking.pdf
- Appendice 2 - Analisi applicativa.pdf