BADAN PENGAWASAN KEUANGAN DAN PEMBANGUNAN (BPKP)

PUSAT INFORMASI PENGAWASAN

INSTRUKSI KERJA
MANAJEMEN INSIDEN KEAMANAN TI

No. Dok	: IK-IP/KTI-01
Tanggal Berlaku	: 27 Maret 2023
No. Rev/Tanggal	: 00/-

Disusun Oleh Koordinator SMKI	Disahkan Oleh Kepala Pusinfowas BPKP
Mohammad Fahmi Kurniawan NIP. 19750510 199502 1 001	Moch. Fahrudin NIP. 19720218 199202 1 001

RIWAYAT REVISI

Revisi	Deskripsi	Diusulkan Oleh	Tanggal
00	Penerbitan dokumen	Tim Keamanan SMKI	

1. TUJUAN

Penanganan yang terencana dan terorganisir sangatlah diperlukan dalam hal terjadinya sebuah insiden, supaya hal tersebut dapat dilakukan, maka diperlukan adanya suatu instruksi kerja dalam menangani insiden tersebut. Secara umum, tujuan instruksi kerja insiden ini adalah sebagai panduan untuk pengelola Teknologi Informasi jika terjadi insiden dan sebagai dokumentasi untuk setiap insiden yang terjadi dalam proses pengelolaan Teknologi Informasi.

2. RUANG LINGKUP

Penanganan insiden keamanan teknologi informasi mencakup insiden yang terjadi pada aset TI di lingkungan BPKP.

3. ACUAN/REFERENSI

3.1 Keputusan Kepala BPKP Nomor HK.01/Kep.461/K/IP/2022 tentang Tim Penanggulangan dan Penanganan Siber (*Computer Security Incident Response Team*)

3.2 RFC 2350 BPKP-CSIRT

3.3 Panduan Penanganan Insiden Keamanan TI BSSN

4. ISTILAH/DEFINISI

Insiden Keamanan adalah proses pengelolaan gangguan keamanan teknologi informasi.

5. PIHAK-PIHAK YANG TERLIBAT

Pihak-pihak yang terlibat di dalam penanggulangan dan penanganan insiden keamanan meliputi tim Pusinfowas dan pihak eksternal Pusinfowas sebagaimana diatur pada Lampiran I Keputusan Kepala BPKP Nomor HK.01/Kep.461/K/IP/2022 tentang Tim Penanggulangan dan Penanganan Siber (*Computer Security Incident Response Team*).

6. Prosedur Response Pelaporan Insiden Keamanan TI

6.1 Laporan Insiden

Laporan Insiden diperoleh secara internal dari tim teknis Pusinfowas ataupun secara eksternal dari tim teknis penghubung. Untuk setiap insiden yang dilaporkan, tim teknis Pusinfowas melaksanakan dokumentasi setiap laporan yang masuk minimal meliputi apa yang terjadi (foto/screenshot), dimana insiden terjadi, kapan insiden diketahui terjadi, waktu pelaporan insiden dan siapa pihak yang melaporkan.

6.2 Verifikasi Laporan

Tim Teknis Keamanan melakukan tindakan:

1. Menginspeksi langsung insiden tersebut sebagai bahan dalam tindak lanjut insiden.
2. Mengkoordinasikan dengan pihak pelapor yang meliputi tim teknis penghubung atau tim teknis Pusinfowas.
3. Tim Teknis Keamanan merumuskan alternatif rencana penanganan insiden.
4. Tim Teknis Keamanan meminta approvement kepada Wakil Ketua Tim Teknis CSIRT.

6.3 *Approvement*

1. Ketua Tim Teknis Keamanan menyampaikan laporan insiden dan alternatif penanganan insiden kepada Wakil Ketua Tim Teknis CSIRT.
2. Wakil Ketua Tim Teknis CSIRT memberikan approvement penanganan insiden.

6.4 Respon Insiden

Melakukan tindakan-tindakan meliputi:

1. Identifikasi
2. Containment
3. Eradication
4. Pemulihan/Recovery
5. Tindak Lanjut

Tindakan-tindakan tersebut harus dikoordinasikan bersama dengan tim teknis Keamanan TI, Tim Teknis Pengelola Jaringan dan Server, Tim Teknis Layanan TI, dan Tim Teknis Penghubung.

Apabila insiden telah selesai ditangani, tim teknis keamanan menyusun laporan hasil penanganan insiden termasuk catatan waktu pemulihan/*recovery.*

7. Response Insiden

7.1 Insiden *Web Defacement*

1. Identifikasi

a. Lakukan pemeriksaan dan analisis pada *log server* (*Error Log, Access Log, Database Log, Auth Log, Install Log, Event Log,Firewall Log, IDS/IPS Log, Switch/Router Log*) untuk menemukan *file* atau aktifitas mencurigakan.

b. Analisa daftar user pada *server* dengan berkoordinasi dengan subkoor PITI.
c. Analisa *History command* pada server dengan command "*history*" atau *open file history.*
d. Lakukan pengecekan *background service* yang berjalan pada *server* aplikasi.

2. *Containment*

a. Lakukan pembangunan *site under maintenance*;
b. Lakukan backup sistem, untuk keperluan forensik ataupun untuk mengumpulkan bukti-bukti insiden;
c. Mematikan *service web server* aplikasi.
d. Melakukan *backup file bash history*.

3. *Eradication*

Setelah ditemukan aplikasi ataupun file yang bersifat *malicious*, maka tahap selanjutnya adalah melakukan penghapusan konten tersebut. Adapun tahapannya adalah sebagai berikut :

a. Lakukan hapus *file malicious*, antara lain : *file defacement, file backdoor, file rootkit ataupun file malware*;
b. Melakukan r*eset password* dan hapus user yang tidak terdaftar pada aplikasi;
c. Lakukan *uninstall* aplikasi yang ditemukan sebagai aplikasi *malicious*;

4. Pemulihan/*Recovery*

a. Mengaktifkan (*restore*) file-file yang telah di-*backup*. File dapat berupa file pada *web server*, *file database* dan gunakan aplikasi *checksum* sebagai *data integrity checker* pada file *backup* tersebut;
b. Lakukan *update/upgrade/patch* semua aplikasi yang digunakan pada web server. Jika menggunakan CMS, update versi web aplikasi, plugins, themes yang digunakan. Jika menggunakan API dapat melakukan *update library* yang digunakan. Selain itu perlu dilakukan update rules pada konfigurasi keamanan yang digunakan;
c. Lakukan *automatic updates* pada setiap aplikasi yang digunakan;
d. Lakukan pembaruan seluruh akun yang digunakan baik pada sistem operasi ataupun web aplikasi;
e. Lakukan *hardening server* ataupun aplikasi yang digunakan seperti memasang *Web Application Firewall* (WAF), memasang aplikasi *anti-defacement* (*DotDefender, Nagios, Webguard*);
f. Melakukan konfigurasi *enable X-Forwarded* pada *format access log* agar dapat mengetahui IP real dari pengakses aplikasi.

5. Tindak Lanjut

Sebagai tindak lanjut penanganan insiden, perlu dilakukan hal-hal sebagai berikut :

a. Melakukan scan VA pada aplikasi yang akan masuk ke server *production*;
b. Memetakan kerentanan yang ditemukan, baik rentan terhadap serangan *SQL Injection, XSS, Misconfiguration*, atau sudah *deprecated*/usangnya versi aplikasi yang digunakan;
c. Membuat semua dokumentasi dan laporan terkait kegiatan dan waktu yang dibutuhkan pada proses *incident handling* yang telah dilakukan;
d. Menuliskan *tools* apa saja yang digunakan dalam membantu proses incident handling;
e. Menuliskan bukti-bukti yang ditemukan, hal ini terkait dengan proses hukum kedepannya;
f. Memberikan analisa dan penjelasan apa yang harus dilakukan sehingga insiden serupa tidak terulang kembali;
g. Membuat evaluasi dan rekomendasi.

7.2 Insiden *Malware*

1. Identifikasi

a. Periksa apakah antivirus berfungsi normal atau tidak.
b. Mengecek file yang tidak dikenal pada *root* atau *system directory*.
c. Periksa file dengan ekstensi ganda. Sangat disarankan untuk menonaktifkan opsi fitur 'sembunyikan ekstensi' pada file explorer untuk mengetahui ekstensi yang sebenarnya dari suatu file.
d. Periksa proses dan *service* yang tidak dikenal dalam sistem menggunakan Task Manager
e. Periksa utilitas sistem, misalnya *Task Manager* atau *SysInternals Process Explorer*. Terdapat *malware* yang menonaktifkan utilitas ini sehingga tidak dapat dijalankan.
f. Periksa penggunaan memory CPU menggunakan *Task Manager*.
g. Periksa *anomali* pada *Registry Key*.
h. Memeriksa *anomali* pada *traffic* jaringan. *Malware modern* saat ini kebanyakan memiliki fitur "*Command and Control*" dimana biasanya setiap *malware* yang sudah menginfeksi suatu sistem, akan mengirimkan sinyal kepada induk *malware* melalui aktivitas "*Command and Control*" tersebut.
i. Identifikasi anomali proses dan *service* yang dibuat pada *Task Scheduler*.

j. Identifikasi *user account* pada sistem. Beberapa malware mempunyai kemampuan untuk membuat *user account* baru pada sistem operasi yang terinfeksi.
k. Identifikasi *entry log* pada sistem operasi menggunakan *Event Viewer*.
l. Identifikasi proses yang mencurigakan menggunakan *SysInternals Tools*. *SysInternal Tools* merupakan salah satu kumpulan *tools* utilitas milik Microsoft yang bertujuan untuk mengidentifikasi sistem lebih mendetail. Beberapa Aplikasi *SysInternal tools* yang paling banyak digunakan untuk melakukan identifikasi dan analisa *malware* adalah *Process Explorer, Autoruns*, dan *Process Monitor*.

2. *Containment*

a. Meminta izin kepada pemilik sistem untuk memutus sistem yang terinfeksi *malware* dari jaringan.
b. Isolasi sistem yang terinfeksi malware. Hal ini dapat dilakukan dengan cara mencabut kabel LAN atau memindahkan sistem tersebut ke VLAN khusus. Namun, perlu menyimpan informasi koneksi jaringan pada sistem sebelum memutuskan hubungan dari jaringan yang mungkin akan dibutuhkan dalam melakukan analisa selanjutnya.
c. Mengubah konfigurasi *routing table* pada *Firewall* untuk memisahkan sistem yang terinfeksi *malware* dengan sistem lainnya.
d. Melakukan backup data pada sistem yang terinfeksi malware.
e. Identifikasi gejala kemiripan pada sistem lain untuk mencegah penyebaran malware. Jika terdapat kemiripan, maka sistem tersebut juga harus dilakukan proses containment.

3. *Eradication*

a. Menghentikan proses yang terindikasi sebagai proses yang *malicious*, dengan cara sebagai berikut :
   I. Tidak melakukan *kill / end proces* terhadap *malicious process* tersebut. Hal ini dikarenakan *malware* akan melakukan *auto start process* ketika prosesnya terhenti.
   II. Lakukan *suspend* terhadap proses tersebut, kemudian lakukan *record* pada *path EXE* proses tersebut dan *file DLL* yang dipanggil oleh proses tersebut.
   III. Dalam kondisi *sleep* (proses di *suspend*), kemudian satu persatu lakukan kill process dari kumpulan malicious process tersebut dimulai dari *child process* ke *parent process*.
   IV. Jika *malicious process* masih melakukan *auto start* atau mengganti Namanya dengan nama proses baru, maka perlu

didokumentasikan lebih lanjut dan simpan *malicious program* tersebut ke media lain untuk proses analisis yang lebih mendetail.

b. Menghapus *auto start process* yang mencurigakan dari hasil analisa aplikasi autostart.
c. Jika proses tersebut kembali lagi, jalankan *Process Monitor* untuk mengidentifikasi apakah ada lokasi lain dimana malware tersebut bersembunyi.
d. Lakukan proses di atas secara berulang hingga dapat dipastikan semua malicious program telah dihapus dan prosesnya sudah di *kill process*.
e. Setelah program *malware* dihapus dan *malicious process* di *kill process*, lakukan *full scanning* terhadap sistem menggunakan *signature antivirus* yang sudah diperbaharui.
f. Jika proses *scanning antivirus* tidak dapat dilakukan karena telah diblokir oleh malware, maka lakukan proses sebagai berikut :
    I. Booting sistem melalui Live usb rescue disk, misalnya Hiren Boot CD, FalconFour's Ultimate Boot CD, Kaspersky Rescue Disk, dll.
        a. Live usb tersebut dapat berupa sistem operasi Linux ataupun miniXP yang berisi beberapa *tools* seperti defragment tools, driver tools, backup dan recover data tools, antivirus dan anti-malware tools, rootkit detection tools, secure data wiping tools, partitioning tools, password recovery tools, network tools, recover/repair broken partitions tools, dll. Lakukan proses mounting sistem operasi yang terinfeksi ke dalam Live usb yang sedang berjalan.
        b. Lakukan proses scanning antivirus dan antimalware pada Live usb yang sedang berjalan
g. Jika terdapat user yang dibuat oleh *malware*, maka hapus user yang tidak dikenali tersebut untuk menghindari masuknya kembali *malware* melalui user yang tidak dikenal tersebut.

4. Pemulihan/*Recovery*

Pemulihan merupakan tahap untuk memulihkan data sistem yang terinfeksi malware serta mengembalikan seluruh sistem bekerja normal seperti semula. Langkah yang dilakukan terhadap pemulihan sistem, diantaranya:

a. Validasi sistem untuk memastikan sudah tidak ada aplikasi atau file yang rusak atau terinfeksi malware. Begitu pula kesalahan atau kekurangan konfigurasi sistem untuk kemudian disesuaikan kembali.
b. Melakukan aktivitas monitoring untuk memastikan apakah malware masih ada atau kembali lagi setelah proses eradication dengan melakukan hal-hal sebagai berikut :
    I. Memantau proses dan *service* yang berjalan menggunakan *Process Monitor* dan *Process Explorer*.
    II. Memantau aktivitas traffic jaringan menggunakan tools wireshark atau tcpdump untuk memantau apakah ada request outgoing atau traffic incoming yang mencurigakan, serta request query DNS karena malware yang memiliki kemampuan Command and Control biasanya melakukan kontak dengan induknya.
c. Jika terjadi kerusakan yang cukup parah (file sistem terhapus, data penting hilang, menyebabkan kegagalan booting pada sistem operasi), maka sistem dibangun ulang dari file backup terakhir sistem yang dimiliki.
d. Melakukan *patching* sistem.
e. Melakukan hardening terhadap sistem.
f. Menambahkan signature dari malware ke sistem monitoring atau database antivirus.

5. Tindak Lanjut

Tahap ini adalah *fase* di mana semua dokumentasi kegiatan yang dilakukan dicatat sebagai referensi untuk masa mendatang. Prosedur yang dapat dilakukan adalah sebagai berikut:

a. Membuat dokumentasi dan laporan terkait penanganan insiden malware, yang berisi langkah-langkah dan hasil yang telah didapatkan.
b. Memberikan analisa dan penjelasan apa yang harus dilakukan, sehingga meminimalisir insiden serupa tidak terulang kembali.
c. Menuliskan bukti-bukti yang ditemukan, hal ini terkait dengan proses hukum kedepannya.
d. Membuat evaluasi dan rekomendasi. Rekomendasi yang dapat diberikan diantaranya:
    I. Penambahan pengetahuan tentang penanganan insiden malware, misalnya melalui pelatihan

II. Memperbaharui anti malware dengan *signature file* yang baru, dengan harapan dapat berhasil dalam mendeteksi dan menghapus malware

III. Meningkatkan pertahanan sistem terhadap malware

e. Mendokumentasikan malware terkait jalan masuk, perilaku, dampak kerusakan, dll yang terkait malware ke dalam *database malware.*

f. Menyempurnakan langkah-langkah respon atau prosedur penanganan insiden malware yang ada.

7.3 Insiden Serangan DDoS

1. Identifikasi

Langkah-langkah yang dapat diambil pada tahap identifikasi dan analisis antara lain:

a. Mengetahui perilaku "normal" dari lalu lintas jaringan, penggunaan CPU, penggunaan memori dari *host*, sehingga alat monitoring jaringan akan memberikan informasi berupa peringatan terhadap perubahan abnormal. Beberapa indikasi bahwa telah terjadi serangan DDoS diantaranya:
   - Melambatnya lalu-lintas jaringan
   - Melambatnya proses pada komputer host
   - Penggunaan ruang disk yang bertambah
   - Layanan tidak dapat diakses atau sistem crash
   - Waktu login yang lama, bahkan ditolak
   - Log penuh
   - Anomali pada fungsi *port*

b. Mengidentifikasi komponen infrastruktur yang terkena dampak.

c. Berkoordinasi dengan pihak terkait untuk mengetahui apakah jaringan organisasi merupakan target utama atau korban dari imbas (misalnya imbas dari serangan terhadap penyedia layanan internet atau penyedia hosting).

d. Memeriksa lalu lintas jaringan, seperti *source IP address*, *destination port*, URLs, protocol, TCP sysnc, UDP, ICMP dan *traffic Netflow* misalnya menggunakan tcpdump, wireshark, snort dan membandingkannya dengan lalu lintas jaringan "normal". Dengan memeriksa lalu lintas jaringan, juga dapat diketahui sumber dan jenis serangan.

e. Menganalisa *file log* yang tersedia (*file log server, router, firewall*, aplikasi dan infrastruktur lainnya yang terkena dampak) untuk mengetahui jenis serangan, sumber serangan, apa yang menjadi sasaran, dan bagaimana masuknya serangan.

f. Menentukan dampak dari tingkat keparahan yang terjadi, yaitu seberapa besar sistem dan layanan mengalami gangguan, serta kemungkinan motif yang dilakukan oleh penyerang.

2. *Containment*

Tahap *containment* bertujuan untuk meminimalisir efek/dampak serangan pada sistem yang ditargetkan dan mencegah kerusakan lebih lanjut. Prosedur yang dilakukan pada tahap ini adalah:

a. Jika sumber *bottleneck* berada pada fitur tertentu dari suatu aplikasi (dalam artian suatu aplikasi sedang menjadi target), maka perlu mempertimbangkan untuk menonaktifkan sementara aplikasi tersebut.
b. Jika *bottleneck* berada di ISP, maka perlu berkoordinasi dengan pihak ISP untuk meminta *filtering*.
c. Merelokasi target ke alamat IP lain jika suatu host tertentu sedang menjadi target (sebagai solusi sementara).
d. Jika memungkinkan, memblokir lalu lintas yang terhubung dengan jaringan (*router, firewall, load balancer*, dll).
e. Mengontrol lalu lintas data dengan menghentikan koneksi atau proses yang tidak diinginkan pada *server/router*.
f. Melakukan *filter* sesuai karakteristik serangan, misalnya memblokir paket *echo* ICMP.
g. Menerapkan *rate limiting* untuk protokol tertentu, mengijinkan dan membatasi jumlah paket per detik untuk protokol tertentu dalam mengakses suatu *host*.

3. *Eradication*

Eradication pada penanganan serangan DDoS yaitu mengambil tindakan untuk menghentikan kondisi d*enial of service*. Tindakan ini sebagian besar melibatkan peran ISP. Prosedur untuk melakukan proses ini dapat dilakukan dengan cara menghubungi penyedia layanan internet (ISP) untuk meminta bantuan, terkait:

a. Pemblokiran jaringan (*source IP address)*
b. Pemfilteran (membatasi jumlah lalu lintas)
c. *Traffic-scrubbing/sinkhole/clean-pipe*
d. *Blackhole routing*

4. **Pemulihan/*Recovery***

   Pemulihan merupakan tahap untuk mengembalikan seluruh sistem bekerja normal seperti semula. Memahami karakteristik serangan diperlukan untuk pemulihan yang cepat dan tepat. Prosedur yang dapat dilakukan pada tahap pemulihan diantaranya sebagai berikut:

   a. Memastikan bahwa serangan DDoS pada jaringan telah selesai dan layanan bisa dilakukan kembali.
   b. Memastikan bahwa jaringan telah kembali ke kinerja semula
   c. Memastikan bahwa layanan yang terkena dampak dapat dijangkau lagi/beroperasi kembali.
   d. Memastikan bahwa infrastruktur telah kembali ke kinerja semula (tidak ada kerusakan)
   e. Memulai layanan, aplikasi dan modul yang ditangguhkan
   f. Mengembalikan ke jaringan asli dan mengalihkan kembali lalu lintas ke jaringan asli.

5. **Tindak Lanjut**

   Pemulihan merupakan tahap untuk mengembalikan seluruh sistem bekerja normal seperti semula. Memahami karakteristik serangan diperlukan untuk pemulihan yang cepat dan tepat. Prosedur yang dapat dilakukan pada tahap pemulihan diantaranya sebagai berikut:

   a. Memastikan bahwa serangan DDoS pada jaringan telah selesai dan layanan bisa dilakukan kembali.
   b. Memastikan bahwa jaringan telah kembali ke kinerja semula
   c. Memastikan bahwa layanan yang terkena dampak dapat dijangkau lagi/beroperasi kembali.
   d. Memastikan bahwa infrastruktur telah kembali ke kinerja semula (tidak ada kerusakan)
   e. Memulai layanan, aplikasi dan modul yang ditangguhkan
   f. Mengembalikan ke jaringan asli dan mengalihkan kembali lalu lintas ke jaringan asli.

7.4 Insiden Serangan *Phising*

1. **Identifikasi**

   Tujuan dari proses identifikasi adalah untuk mendeteksi adanya insiden serangan phising, menentukan ruang lingkup, dan melibatkan pihak-pihak yang tepat dalam menangani serangan phising. Tahap identifikasi penanganan serangan phishing adalah sebagai berikut:

   a. Memonitor *email*, *social media*, *web forms* dsb pada Organisasi untuk mencari informasi *Phising*;

b. Memeriksa URL *phising* dan *hyperlink* yang mencurigakan menggunakan www.virustotal.com, www.urlvoid.com, serta www.phishtank.com;
c. Melibatkan pihak yang tepat terkait serangan *phising*. Agar bisa segera dilakukan *takedown* terhadap *web phising*. Seperti perusahaan *hosting*, penyedia domain, penyedia jasa email, Nasional CSIRT;
d. Mengumpulkan bukti bukti terkait adanya serangan *phising*. Contohnya *screenshot* halaman *web* yang terdampak.

2. ***Containment***

Setelah dipastikan bahwa memang benar telah terjadi serangan *phishing*, maka dilakukan proses mitigasi serangan, agar tidak terjadi kerusakan lebih dalam. Prosedur yang dilakukan pada tahap ini adalah:

a. Menyebarkan URL *phising* dan konten dari *email phising* pada pihak *spam-reporting website*, misalnya www.phishtank.com;
b. Menginformasikan serangan *phising* kepada pengguna, agar pengguna mengetahui dan tidak terkena dampak dari serangan tersebut;
c. Memeriksa *source code* dari *website phising*, jika menggunakan gambar dari *website* yang anda miliki, anda dapat mengganti gambar dengan tampilan "*PHISING WEBSITE*".

3. ***Eradication***

Proses ini bertujuan untuk mengambil tindakan dalam menghentikan serangan *phising*. Prosedur untuk melakukan proses ini dapat dilakukan dengan cara berikut:

a. Jika halaman *phising* di *hosting* di situs web yang telah disusupi, maka hubungi pemilik dari *website* tersebut, agar halaman *phising* dihapus dan dilakukan *update security*;
b. Untuk percepatan penanganan, hubungi perusahaan *hosting* dengan mengirim *email* berisikan informasi phising, serta lakukan kontak telepon perusahaan hosting yang tersedia;
c. Menghubungi perusahaan hosting untuk melakukan *takedown* /penutupan alamat *website* palsu;
d. Jika *takedown* terlalu lama, maka hubungi Nasional CSIRT untuk membantu proses *takedown*.

4. **Pemulihan/*Recovery***

   Pemulihan merupakan tahap untuk mengembalikan seluruh sistem bekerja normal seperti semula. Prosedur yang dapat dilakukan sebagai berikut:

   a. Memastikan bahwa halaman *website* penipuan sudah tidak dapat diakses;
   b. Tetap Memantau URL palsu, untuk memastikan URL palsu tersebut tidak dapat diakses;
   c. Menghapus halaman peringatan dari *website*.

5. **Tindak Lanjut**

   Prosedur yang dapat dilakukan adalah sebagai berikut:

   a. Menyempurnakan langkah-langkah respon, prosedur penanganan serangan yang diambil selama insiden agar kedepannya dapat menangani insiden secara lebih cepat dan efisien;
   b. Memperbarui daftar kontak yang dimiliki, disertai catatan cara paling efektif untuk menghubungi setiap pihak yang terlibat;
   c. Berkolaborasi dengan tim hukum jika diperlukan tindakan hukum;
   d. Membuat dokumentasi dan laporan terkait penanganan serangan Phising;
   e. Membuat evaluasi dan rekomendasi.

7.5 Insiden Serangan SQL *Injection*

1. **Identifikasi**

   a. Memeriksa alert dan anomalies dari perangkat IDS atau IPS;
   b. Melakukan *error checking* melalui *form* atau *url* dengan memberikan karakter atau sebuah simbol. Misalnya:
   c. Memeriksa semua log (*error log, access log, database log, firewall log*). Lokasi *log file* secara default berada pada *var/log*, *log* tersebut menyimpan seluruh aktivitas yang terjadi pada sistem;
   d. Memeriksa adanya *command line*, *string-string* yang digunakan untuk menyerang;
   e. Memeriksa isi *database* untuk mencari *script* yang berbahaya, dan mengecek apakah ada penambahan user secara tidak sah;
   f. Memeriksa apakah ada file atau script berbahaya (*trojan*, *malicious file*, *backdoor*) yang ditanamkan pada web server;
   g. Menggunakan *tool* untuk memeriksa kerentanan. *Tool* yang dapat digunakan diantaranya *Acunetix, SQLMap, SQL Injection tools.*

2. **Containment**
   a. Melakukan proses backup semua data yang terdapat pada web server. untuk keperluan forensik dan pengumpulan bukti-bukti. Backup sebaiknya ditempatkan pada hard disk eksternal;
   b. Jika sumber penyerangan berasal dari sistem lain pada jaringan, maka putuskan secara fisik koneksi tersebut dan lakukan investigasi sumber tersebut.

3. ***Eradication***
   Tahap Eradication pada penanganan serangan SQL Injection adalah untuk menghapus *file/script* serta menutup sumber serangan. Prosedur untuk melakukan proses ini dapat dilakukan dengan cara berikut:
   a. Memeriksa apakah terdapat *malicious file*, *backdoor*, *rootkit* atau kode-kode berbahaya lainnya yang berhasil ditanamkan pada server dan segera menghapusnya;
   b. Jika terdapat kode SQL yang mengakses IP tertentu maka perlu melakukan block/menutup sumber serangan (*block* IP dan *Port*).

4. **Pemulihan/*Recovery***
   Pemulihan merupakan tahap untuk mengembalikan seluruh sistem bekerja normal seperti semula. Prosedur yang dapat dilakukan sebagai berikut:
   a. Mengubah credential password pengguna. Hal ini untuk mengantisipasi apabila *password* pengguna telah diketahui oleh penyerang;
   b. Melakukan *recovery database* pada aplikasi web;
   c. Jika *SQL Injection* menyebabkan web defacement, gunakan panduan penanganan insiden *web defacement*;
   d. Jika *SQL Injection* menyebabkan insiden *malware*, gunakan panduan penanganan insiden *malware*;
   e. Menutup semua kerentanan yang telah diketahui;
   f. Membatasi akses *root* langsung ke *database*;
   g. Melakukan filter terhadap input yang dimasukkan oleh pengguna;
   h. Mematikan atau menyembunyikan pesan-pesan *error* yang keluar dari *SQL Server* yang berjalan;
   i. Patching terhadap aplikasi yang rentan, melakukan upgrade terhadap aplikasi web yang masih memiliki kerentanan;
   j. Melakukan *penetration testing* untuk mengetahui celah-celah keamanan yang mungkin masih terdapat pada *website*.

5. Tindak Lanjut

Prosedur yang dapat dilakukan adalah sebagai berikut:

a. Membuat dokumentasi dan laporan terkait penanganan serangan *SQL Injection*;
b. Menuliskan *tools* apa saja yang digunakan dalam penanganan serangan injeksi sql;
c. Menuliskan bukti-bukti yang ditemukan, hal ini terkait dengan proses hukum kedepannya;
d. Memberikan analisa dan penjelasan apa yang harus dilakukan sehingga serangan serupa tidak terulang kembali;
e. Membuat evaluasi dan rekomendasi.