Framework Verifikasi Keamanan DNS v1.0

Girindro Pringgo Digdo

DAFTAR ISI

Pengantar

Saat ini saya baru saja menyelesaikan sebuah *project* kecil-kecilan. *Project* ini adalah sebuah dokumen *framework* yang dapat digunakan untuk melakukan verifikasi keamanan terhadap DNS.

Framework ini dibuat dengan tujuan agar memudahkan para teknisi untuk melakukan pengecekan terhadap konfigurasi DNS yang akan dan telah mereka implementasikan. *Framework* ini saya buat berdasarkan referensi dari *National Institute of Standards and Technology* (NIST) SP800-81-2.

Adapun istilah yang ada dalam dokumen ini sengaja tidak saya terjemahkan ke dalam Bahasa, guna menghindari kesalahpahaman dalam memahaminya.

Dokumen ini memiliki versi 1.0 yang artinya dokumen ini dapat diperbaiki, diubah serta disempurnakan sesuai dengan masukan dari pembaca dan perkembangan teknologi nantinya.

Oleh karena itu, kontribusi pembaca dalam mengoreksi dokumen ini dibutuhkan sehingga dapat menjadi dokumen acuan dalam melakukan verifikasi keamanan DNS, khususnya di Indonesia.

Akhirnya, mohon maaf apabila bahasa yang digunakan dalam dokumen ini masih sulit dipahami.

Semoga bermanfaat :)

I. Pendahuluan

I.1 Latar Belakang

Domain Name System (DNS) merupakan sebuah mekanisme untuk penamaan terhadap sumber daya (*resources*) yang digunakan pada jaringan global. Mekanisme ini bertujuan agar pengguna dapat dengan mudah mengakses alamat sumber daya yang berlokasi di internet.

Laporan IDSIRTII[1] di tahun 2014 hingga awal tahun 2016 menyebutkan bahwa *traffic* serangan didominasi oleh DNS. Serangan ini merupakan serangan terbesar yang diarahkan ke port 53 dengan total mencapai 29 ribu serangan atau seribu serangan per hari.

Port	Total	Perhari
53	29,644	956
4444	14,822	478
1434	12,567	405
80	9,716	313
4500	5,865	189
16512	4,342	140
8198	3,018	97
8195	2,004	64
8204	1,929	62
8196	1,927	62
Total	85,834	2,769

Tabel 1. Statistik Serangan pada Januari 2016

Dari tabel 1 di atas terlihat bahwa port 53 (DNS) merupakan port yang paling banyak terkena serangan yaitu sebesar 29.644 serangan. Hal ini mengindikasikan bahwa tren serangan yang terjadi pada dua tahun terakhir menempatkan DNS pada urutan pertama.

NIST SP800-81-2[2] merupakan dokumen standar yang menyediakan panduan dalam mengimplementasikan pengamanan DNS. Tujuan utama pengamanan ini adalah integritas data (*data integrity*) dan sumber otentikasi (*source authentication*). Hal tersebut dibutuhkan untuk menjamin ke-otentikan terhadap

sebuah informasi nama domain, serta memelihara integritas terhadap informasi nama domain ketika melintasi jaringan. Standar ini juga menyediakan panduan yang lengkap dalam memelihara integritas data dan melakukan otentikasi sumber.

Framework ini memetakan hal-hal apa saja yang menjadi komponen dalam melakukan verifikasi keamanan DNS. Diharapkan dengan adanya *framework* ini, seorang teknisi dapat dengan mandiri menilai tingkat keamanan DNS dengan melakukan verifikasi terhadap kontrol yang telah diterapkan. Hasil verifikasi tersebut dapat dijadikan acuan dalam melakukan tindakan perbaikan keamanan.

I.2 Tujuan

Tujuan umum dari *framework* ini adalah untuk merancang kerangka kerja verifikasi keamanan DNS. Sedangkan tujuan khusus yang ingin dicapai dalam *framework* ini adalah sebagai berikut:

1. Menyediakan *framework* sebagai alat ukur dalam menilai tingkat keamanan DNS.
2. Sebagai pedoman kepada teknisi tentang kontrol keamanan apa saja yang harus dipertimbangkan dalam mengimplementasikan DNS.
3. Sebagai acuan untuk melakukan tindak lanjut perbaikan keamanan

I.3 Batasan Masalah

1. *Framework* ini mengacu kepada NIST SP800-81-2.
2. *Framework* ini digunakan hanya untuk menilai keamanan DNS.
3. Verifikasi keamanan dilakukan dengan berfokus pada *data integrity* dan *source authentication.*
4. Verifikasi keamanan tidak termasuk melakuan evaluasi terhadap Orang/Personil, Proses, dan Administratif (Kebijakan dan Prosedur).

II. *Checklist* Verifikasi Keamanan DNS

II.1 Komponen Verifikasi

Verifikasi dilakukan dengan mengacu kepada komponen yang ada pada NIST SP800-81-2. Komponen tersebut dapat digambarkan sebagai berikut.

Gambar 1. Komponen verifikasi DNS

II.1.1 DNS Hosting Environment

DNS Hosting Environtment merupakan komponen yang mendukung berjalannya DNS pada suatu lingkungan server. Komponen ini diklasifikasikan sebagai berikut:

- *DNS host platform*
- *DNS software*
- *Content control of zone file*

II.1.2 DNS Transactions

DNS Transactions adalah aktivitas suatu DNS yang terjadi di internet. Komponen ini diklasifikasikan sebagai berikut:

- *Restricting Transaction Entities Based on IP Address*
- *Transaction Protection through Hash-Based Message Authentication Codes (TSIG Specification)*

- *Transaction Protection through Asymmetric Digital Signatures (DNSSEC Specification)*

II.1.3 DNS Query/Response

DNS Query/Response merupakan tipe dari sebuah transaksi pada DNS, yaitu klien mengirim DNS kueri, dan server akan membalas (*reply*) melalui *DNS Response.* Perlindungan komponen ini diklasifikasikan sebagai berikut:

- *Data origin authentication*
- *Data integrity verification*
- *Authenticated denial of existence capabilities*

II.1.4 Minimizing Information Exposure Through DNS Data Content Control

Komponen ini akan meminimalisasi informasi yang dapat digunakan oleh penyerang dalam melakukan serangan terhadap organisasi. Informasi yang tersedia pada jaringan harus dijaga dari pihak yang tidak berkepentingan, khususnya informasi terkait dengan konfigurasi pada DNS.

II.1.5 DNS Security Administration Operations

Komponen ini bertanggungjawab terhadap implementasi fitur DNSSEC dalam melindungi transaksi *DNS query/response.* Komponen ini juga bertanggungjawab terhadap administrasi keamanan secara berkala dan termasuk *checklist* yang ada di dalamnya.

II.1.6 Recursive Servers (Resolver) & Stub Resolvers

DNS memiliki dua komponen dasar yaitu *Authoritative name server*, yang berfungsi untuk menerbitkan *DNS data*, dan *DNS resolvers* yang berfungsi menerbitkan kueri untuk *DNS data*. *Resolvers* dapat dibagi menjadi dua yaitu *Stub resolvers* (biasanya ditemukan di *individual hosts*) yang berfungsi menerbitkan kueri, tetapi tidak mengikuti *DNS referral*, dan *Recursive servers/resolvers* yang mengikuti *DNS referral.*

II.1.7 Validating Resolvers

Komponen ini akan melakukan validasi yang terdiri dari:

- *Data origin authentication*
- *Integrity protection*

II.2 Verifikasi DNS Hosting Environment

II.2.1 *Control Objectives*

Memastikan bahwa *DNS Hosting Enviroment* telah melakukan pengamanan dari aspek:

- *DNS Host Platform*
- *DNS Software*
- *Content Control of Zone File*

II.2.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
1.1	Ketika menginstal versi baru dari perangkat lunak *name server*, administrator harus membuat perubahan yang diperlukan terhadap parameter konfigurasi, dengan tujuan mendapatkan fitur keamanan yang baru.		
1.2	Administrator harus menyadari kerentanan, eksploitasi, perbaikan keamanan dan *patch*, baik dari versi terbaru maupun versi sebelumnya yang digunakan di perusahaan.		
1.3	Untuk mencegah informasi tentang versi perangkat lunak yang berjalan pada sistem, *name server* harus dikonfigurasi agar menolak ketika ada permintaan terkait informasi versi.		
1.4	*Authoritative name server* harus tersebar baik jaringan maupun secara geografis. Pada jaringan, dapat dipastikan dengan semua *name server* tidak berada di belakang satu *router* atau *switch*, di dalam satu subnet, atau menggunakan satu jalur *leased line*. Untuk geografis, dapat dipastikan dengan tidak semua *name server* berlokasi di tempat		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
	yang sama, dan minimal terdapat *server backup off-site.*		
1.5	Jika *hidden master* digunakan, *server hidden authoritative master* hanya harus menerima permintaan transfer zona dari kumpulan server di zona sekunder, dan menolak semua permintaan DNS lain. Alamat IP dari *hidden master* harus tersembunyi dalam *name server* yang ada di *database*.		
1.6	Untuk implementasi split DNS, harus ada minimal dua *file* fisik. Satu harus secara eksklusif menyediakan *name resolution* untuk *host* yang terletak di dalam *firewall*. Hal ini juga dapat berisi RRsets untuk *host* di luar *firewall*. *File* yang lain juga menampikan *name resolution*, hanya untuk host yang berlokasi di luar *firewall* atau di DMZ, dan bukan untuk *host* dalam *firewall*.		

II.3 Verifikasi DNS Transactions

II.3.1 *Control Objectives*

Memastikan bahwa *DNS Transactions* telah melakukan pengamanan dari aspek:

- *Restricting Transaction Entities Based on IP Address*
- *Transaction Protection through Hash-Based Message Authentication Codes (TSIG Specification)*
- *Transaction Protection through Asymmetric Digital Signatures (DNSSEC Specification)*

II.3.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
2.1	Panjang kunci TSIG (*Transaction SIGnature*) harus minimal 112 bit jika utilitas pembangkit telah terbukti menghasilkan *string* yang cukup acak (mengacu ke NIST SP800-57P1), atau jika tidak, rekomendasi panjang kunci adalah 128 bit.		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
2.2	Kunci unik TSIG harus dihasilkan untuk setiap pasangan *host* yang berkomunikasi (contoh: kunci terpisah untuk setiap secondary name server untuk mengontentikasi transaksi dengan *primary name server*, dsb).		
2.3	Setelah *key string* disalin ke *key file* dalam *name server*, dua *file* yang dihasilkan oleh *DNSSEC-Keygen* harus dapat diakses dengan akun *administrator* (contoh: *root* di Unix) atau lebih baik lagi, *file* tersebut dihapus. Salinan *file* tersebut juga harus dihancurkan.		
2.4	*Key file* harus ditransmisikan secara aman melalui jaringan menuju *name server* yang akan berkomunikasi dengan *name server* yang menghasilkan *key*.		
2.5	*Key string* harus definisikan dalam *key file* terpisah dan direferensikan melalui *file* konfigurasi. Setiap kunci TSIG harus memiliki *key file* terpisah.		
2.6	*Key file* harus dimiliki oleh akun dimana *software name server* tersebut berjalan. *Permission bit* harus dikonfigurasi sehingga *key file* hanya dapat dibaca atau dimodifikasi oleh akun yang menjalankan *software name server*.		

II.4 Verifikasi DNS Query/Response

II.4.1 *Control Objectives*

Memastikan bahwa *DNS Query/Response* telah melakukan pengamanan dari aspek:

- *Data Origin Authentication*
- *Data Integrity Verification*
- *Authenticated Denial of Existence Capabilities*

II.4.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
3.1	*Name server* yang men-*deploy* DNSSEC harus terkonfigurasi untuk melakukan pengolahan DNSSEC.		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
3.2	*Private key* yang berhubungan dengan ZSK (*Zone-Signing Key*) dan KSK (*Key Signing Key*) tidak harus disimpan pada DNSSEC *primary authoritative name server* ketika *name server* tidak mendukung pembaruan secara otomatis. Namun jika mendukung pembaruan otomatis, maka *private key* tersebut harus disimpan dalam *name server*, dengan dengan direktori / *file-level* yang sesuai dengan kontrol akses atau dilindungi dengan kriptografi.		
3.3	Pembangkitan tanda tangan menggunakan KSK harus dilakukan secara *offline*, menggunakan *KSK private stored offline*; kemudian DNSKEY RRSet, bersama dengan RRSIG RR dapat dimuat ke dalam *primary authoritative name server*.		

II.5 Verifikasi Minimizing Information Exposure Through DNS Data Content Control

II.5.1 *Control Objectives*

Memastikan bahwa Informasi sensitif dari *DNS Data Content Control* telah dibatasi. Aspek tersebut adalah:

- *Parameter Values in SOA RR*
- *Information Leakage from Informational RRTypes*
- *RRSIG Validity Periods*
- *Hashed Authenticated Denial of Existance*
- *Resource Record TTL Value*

II.5.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
4.1	*Refresh value* di zona SOA RR harus dipilih dengan frekuensi pembaruan (*update*). Jika zona telah ditandatangani, *refresh value* harus kurang dari masa berlaku RRSIG.		
4.2	*Retry value* di zona SOA RR harus 1/10 dari *refresh value*.		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
4.3	*Expire value* di zona SOA RR harus 2 hingga 4 minggu.		
4.4	*Minimum TTL (Time To Live) value* harus antara 30 menit dan 5 hari.		
4.5	Adminsitrator DNS harus berhati-hati saat memasukkan HINFO, RP, LOC atau tipe RR lainnya yang dapat membocorkan informasi yang dapat berguna bagi penyerang.		
4.6	Masa berlaku untuk RRSIGs meliputi zona DNSKEY RRset harus dalam kisaran 2 hari untuk 1 minggu. Nilai ini membantu mengurangi periode kerentanan yang dihasilkan dari *key compromise*.		
4.7	Sebuah zona yang memiliki *child* harus memiliki masa berlaku beberapa hari hingga 1 minggu untuk RRSIGs yang meliputi DS RR untuk yang memiliki *child*. Nilai ini membantu mengurangi periode kerentanan pada *zona child* yang dihasilkan dari *KSK compromise* dan *scheduled key rollovers*.		
4.8	Jika zona telah ditandatangani menggunakan NSEC3 RRs, maka nilai *salt* harus diganti setiap waktu setelah zona benar-benar *resigned*. Nilai *salt* harus acak, dan panjangnya harus cukup untuk mencegah FQDN (*fully qualified domain name*) yang terlalu lama bagi protokol DNS (contoh: di bawah 256 oktet).		
4.9	Jika zona telah ditandatangani menggunakan NSEC3 RRs, maka nilai iterasi harus didasarkan pada daya komputasi yang tersedia untuk klien dan penyerang. Nilai tersebut harus ditinjau setiap tahun dan meningkat jika kondisi evaluasi berubah.		
4.10	Nilai TTL untuk DNS data harus diantara 30 menit (1800 detik) dan 24 jam (1440 detik).		
4.11	Nilai TTL untuk RRsets harus menjadi bagian dari masa berlaku tanda tangan pada DNSSEC.		

II.6 Verifikasi DNS Security Administration Operations

II.6.1 *Control Objectives*

Memastikan bahwa *DNS Security Administrations Operations* telah melakukan pengamanan dari aspek:

- *Organizational Key Management*
- *Scheduled Key Rollovers (Key Lifetimes)*
- *Emergency Key Rollovers*
- *Re-Signing a Zone*
- *DNSSEC Algorithm Migration*
- *Special Consideration When Transitioning from NSEC Signed Zones to NSEC3 Signed Zones*
- *DNSSEC in a Split-Zone Deployments*

II.6.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
5.1	Frekuensi *rollover* yang direkomendasikan untuk KSK yaitu sekali setiap 1 hingga 2 tahun, dimana ZSK harus *rolled over* setiap 1 hingga 3 bulan untuk konsistensi operasional.		
5.2	Zona yang belum mempublikasikan kunci publik baru, harus memperhatikan hal-hal berikut:		
5.2.1	Zona aman dimana sebelum mempublikasikan kunci publiknya, harus melakukan setidaknya satu periode TTL sebelum waktu dari *key rollover*.		
5.2.2	Setelah mengapus kunci publik yang lama, zona tersebut harus membangkitkan tanda tangan yang baru (RRSIG RR), berdasarkan tombol yang tersisa (DNSKEY RRs) di dalam *zone file*.		
5.3	Administrator DNS harus memiliki informasi kontak darurat untuk *parent zone*. Hal ini digunakan ketika keadaan darurat *KSK rollover* harus dilakukan.		
5.4	*Parent zone* harus memiliki mekanisme kontak darurat untuk setiap *child subzones* yang didelegasikan.		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
5.5	Periode penandatanganan ulang harus dijadwalkan sebelum berakhirnya RRSIG RRs yang dapat ditemukan dalam zona tersebut. Hal ini untuk mengurangi risiko dari *signed zone* palsu dikarenakan tanda tangan telah kadaluarsa.		
5.6	Nomor seri dalam SOA RR harus bertambah sebelum penandatanganan ulang pada *zone file*.		

II.7 Verifikasi Recursive Servers (Resolver) & Stub Resolvers

II.7.1 *Control Objectives*

Memastikan bahwa *Recursive Servers (Resolvers) & Stub Resolvers* telah melakukan pengamanan dari aspek:

- *Host Platform*
- *Aggregate Caches*
- *Root Hints File*

II.7.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
6.1	*Recursive servers/resolvers* harus ditempatkan di belakang *firewall* dan terkonfigurasi hanya dapat menerima kueri dari *internal hosts* (contoh: *Stub Resolver Host*).		
6.2	Kapanpun *aggregate caches* di-*deploy*, maka *forwarder* harus dikonfigurasi untuk memvalidasi *resolvers.*		
6.3	Setiap *recursive server* harus memiliki *root hints file* yang berisi *IP address* dari satu atau lebih *DNS root servers*. Informasi kebenaran dalam *root hints file* harus dicek secara periodik.		
6.4	*Root hints file* harus dimiliki oleh akun dimana *software name server* berjalan. Akses permission bit harus dikonfigurasi sehingga *root hints file* hanya dapat dibaca atau dimodifikasi oleh akun dimana *software name server* berjalan.		

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
6.5	*Administrator* harus mengkonfigurasi dua atau lebih *recursive resolvers* untuk setiap *stub resolver* yang ada di jaringan.		
6.6	*Enterprise firewall* harus membatasi trafik *DNS outbond* dari *stub resolvers* mengarah ke *recursive resolver* yang dituju.		

II.8 Verifikasi Validating Resolvers

II.8.1 *Control Objectives*

Memastikan bahwa *Validating Resolvers* telah melakukan pengamanan dari aspek:

- *DNSSEC Validation*
- *Establishing Initial Trust Anchors*
- *Maintaining Trust Anchors*

II.8.2 *Requirements*

Ref.	Komponen	Terpenuhi (Ya/Tidak)	Keterangan
7.1	*Non-validating stub resolvers* harus memiliki *trusted link* dengan memvalidasi *recursive resolver*.		
7.2	*Validators* harus secara rutin mencatat kegagalan validasi untuk membantu memeriksa kesalahan jaringan.		
7.3	*Mobile* atau sistem yang nomaden, harus melakukan validasi tersendiri yang telah dipercaya oleh *validator*.		
7.4	*Mobile* atau sistem yang nomaden yang melakukan validasi tersendiri, harus mempunyai kebijakan DNSSEC yang sama dan *trust anchor* sebagai *validator* dalam jaringan organisasi.		
7.5	*Administrator validator* harus mengkonfigurasi satu atau lebih *trust anchors* untuk setiap *validator* dalam organisasi.		
7.6	*Administrator validator* secara perodik harus memeriksa setiap *trust anchor* untuk menjamin bahwa itu masih digunakan, dan melakukan pembaruan *trust anchor* yang diperlukan.		

DAFTAR PUSTAKA

[1] IDSIRTII. ‘Laporan Aktivitas Port per Januari 2016’. 2016.

[2] R. Chandramouli and S. Rose, “Secure domain name system (DNS) deployment guide,” NIST Spec. Publ., 2013.