라우터 보안설정 점검 도구(RAT)

2002년7월 23일

정현철 (hcjung@certcc.or.kr)

1. 개요

라우터의 자체 보안과 네트워크 보호를 위한 여러 가지 설정방법에 대해 "라우터 보안위협요소 및 보안대책"에서 알아보았다.

라우터 보안 위협요소와 보안대책 : http://www.certcc.or.kr/paper/tr2002/tr2002_06/020603-router_security.pdf

하지만, 라우터에서 제공되고 있는 수많은 보안설정 값들을 수동으로 점검하는 것은 정확하지 않을 수 있고, 기관내의 많은 라우터들을 일일이 점검하는 현실적으로 대단히 힘든 일이다.

본 고에서는 다수의 라우터를 대상으로 설정된 환경파일을 자동으로 점검할 수 있는 도구인 RAT(Router Audit Tool)를 소개하도록 한다.

RAT는 Cisco IOS 라우터 환경설정에서 보안 설정상태를 점검하는 도구이다. RAT는 주어진 각각의 라우터에서 보안 문제점을 발견해서 이를 대략적인 보안점수와 함께 리포트 한다. 또한 발견된 문제점에 대해서 hyper link를 통해 문제점에 대한 내용을 자세히 소개하고, 이를 fix할 수 있는 방법을 소개한다.

RAT에서 점검하는 항목들은 NSA의 "Router Security Configuration Guide"에서 언급된 보안설정 Checklist에 적합하게 각 라우터의 설정이 정상적으로 취해졌는지를 자동화하여 점검할 수 있도록 구현하였다.

RAT는 Cisco IOS 버전 11.x와 12.x에서 테스트되어 졌으며 점검방법은 text 형태의 라우터 환경파일을 RAT의 rule과 비교하는 방식으로 이루어진다.

RAT은 다음과 같은 5개의 Perl 프로그램으로 구성되어 있다.

프로그램 명	기능
snarf	환경설정 파일을 가져옴
ncat	rule과 환경설정파일을 읽고 CSV 형태로 출력
ncat_report	CSV 형태의 파일을 읽고 HTML 형태로 출력
rat	실제 점검자가 구동하는 프로그램으로 다른 프로그램들을 실행시킴
ncat_config	각 점검 라우터의 특성에 맞도록 rule을 구성

점검내용은 "services", "access", "logging", "routing" 등 4개의 카테고리로 나눌 수 있으며, 점검 수준은 "Level 1"과 "Level 2"로 나눌 수 있다. "Level 1"은 모든 IOS 설정파일에서 적용가능한 기본적인 보안설정 상태를 점검하는 것이며, "Level 2"는 AAA 인증서버와 같은 각 기관마다 선택적인 점검항목들이 포함된다.

IOS 11.x에 특화된 5개의 점검항목과 IOS 12.x에 특화된 4개 항목을 포함해서 총 66개의 rule을 검사한다. 각 수준별 점검항목은 다음과 같다.

Level 1	Level 2

Apply egress filter	aaa accounting commands
Apply ingress filter	aaa accounting connection
Apply telnet ACL	aaa accounting exec
clock timezone	aaa accounting network
Define telnet ACL	aaa accounting system
disable aux	aaa authentication enable
egress filter definition	aaa authentication login
enable logging	aaa new-model
enable secret	aaa source-interface
encrypt passwords	Apply egress filter to 2nd IF
exec-timeout	Apply ingress filter to 2nd IF
forbid SNMP community private	forbid clock summer-time - GMT
forbid SNMP community public	Loopback0 must be only loopback
forbid SNMP read-write	Loopback0 must exist
forbid SNMP without ACLs	no local logins
ingress filter definition	no tftp-server
logging buffered	ntp server 2
logging console critical	ntp server 3
logging trap debugging	require clock summer-time - localtime
login	require external IF 2 to exist
no cdp run	service timestamps - GMT
no ip bootp server	service timestamps - localtime
no ip http server	tftp source-interface
no ip proxy-arp	Tunnel interfaces must not exist
no ip source-route	
no service config	
no snmp-server	
ntp server	
ntp source	
require external IF to exist	
require line passwords	
set syslog server	
vty transport telnet	
no identd service(IOS 11)	
no directed broadcast(IOS 11, 12)	
no finger service(IOS 11, 12)	
no tcp-small-servers(IOS 11, 12)	
no udp-small-servers(IOS 11, 12)	

2. 설치

RAT는 유닉스 시스템과 윈도우즈 시스템에서 설치 가능하며 본 고에서는 리눅스 시스템에서 설치하기로 한다. RAT는 perl 프로그램으로 구성되어 있으므로 설치에 앞서 Perl 5.004 이상이 설치되어 있어야 한다. 그리고, 라우터 설정 파일(running-config 또는 configuration) 입수는 RAT의 프로그램들 중의 하나인 snarf를 이용할 수도 있지만, Cisco 라우터의 명령어인 "copy run tftp" 등을 이용하여 입수하는 것을 권장한다.

설치는 /usr과 같은 시스템 디렉토리에 설치되어야 하므로 root 권한으로 설치한다.

① www.cisecurity.org/bench_cisco.html에서 최신 버전(rat-1.1.tar.gz)을 다운로드 받는다.(이름, E-mail 주소 등 간단한 정보입력 후 다운로드 가능)

② 압축을 푼다.

```
# gunzip rat-1.1.tar.gz
# tar -xvf rat-1.1.tar
# cd rat-1.1
```

③ 다음 명령어들을 이용하여 설치한다.

```
# perl Makefile-nosnarf.PL [PREFIX=/home/you]
# make
# make test
# make install
```

④ 점검 rule을 자신의 네트워크에 최적화시킨다.
"ncat_config"을 실행하여 로컬 주소, 인터페이스명, 외부망 라우터 여부 등 자신의 환경에 최적화된 환경을 설정한다. 이 결과로 실제 점검 rule 파일인 /usr/etc/ncat.conf가 생성된다.
ncat_config 프로그램은 rule default 파일인 ncat.conf.MASTER 파일을 사용자가 선택한 값으로 바꾸어 /usr/etc/ncat.conf 파일을 생성한다.
네트워크와 라우터 설정에 대한 각각의 질문에 답하고 잘 모를 경우 "?"를 치면 간단한 설명이 주어지며, 그냥 "엔터"를 칠 경우 default 값으로 설정된다.
이 과정은 수십개의 질문을 던져 귀찮게 느껴질 수도 있지만 충실히 자신의 환경을 입력할 수록 보다 정확한 점검이 이루어질 수 있음을 명심하여야 한다.

```
[root@hcjung rat-1.1]# ncat_config
ncat_config: Reading /usr/etc/ncat.conf.MASTER

Please answer the questions below about your network and
router configuration. Type ? to get a short explanation of
any parameter. If you are unsure about what value to give
for a parameter, hit RETURN to take the default value.

Select types of optional rules to be applied:

ncat_config: Apply rules for class use_multiple_ntp_servers [no] ?
ncat_config: Apply rules for class use_multiple_ntp_servers [no] ?

ncat_config: Apply rules for class exterior_router [no] ? ?
Help for class :
An exterior router is a router that connects "internal" networks
such
as desktop networks, DMZ networks, etc to "external" networks
such as the Internet. If this group is chosen, then ingress
and egress filters will be applied to the external interface
See "Building Internet Firewalls" by
Zwicky, Cooper and Chapman, O'Rielly and Associates

Default value is: no

ncat_config: Apply rules for class exterior_router [no] ? yes
...
ncat_config: Enter value for
local_address_internal_netblock_with_mask [192.168.1.0
0.0.0.255] ? 172.16.0.0 0.0.255.255
ncat_config: Enter value for local_address_loopback
[192.168.1.3] ? 172.16.14.2
ncat_config: Enter value for local_timezone [GMT] ?

ncat_config: Writing /usr/etc/ncat.conf...Done.

ncat_config: Now examine /usr/etc/ncat.conf.
ncat_config: Edit /usr/etc/ncat.conf.MASTER and rerun
```

```
ncat_config if not satisfactory.
[root@hcjung rat-1.1]#
```

3. 실행

① 이제 점검결과가 위치하고자하는 디렉토리를 만들고 실행한다.
필자는 Apache 웹서버의 홈디렉토리 아래에 rat이라는 디렉토리를 만들어서 웹 인터페이스를 통해 결과를 확인할 수 있도록 하였다.(최종 점검결과가 HTML 형태로 생성되므로 웹 브라우져를 통해 확인할 수 있도록 웹서버 홈디렉토리 아래에 설치를 권고함)

```
# mkdir /usr/local/apache/htdocs/rat
# cd /usr/local/apache/htdocs/rat
```

② 점검하고자 하는 라우터의 환경파일(텍스트 형태)를 가져온다.
이는 라우터에서 "show running-config" 또는 "show configuration" 명령의 결과를 copy할 수도 있고, 아래와 같이 tftp를 이용하여 받을 수도 있다(tftp를 이용한 파일송수신은 Cisco 문서를 참고바람).

```
router3620#copy run tftp
Address or name of remote host []? 172.16.5.16
Destination filename [router3620-confg]? router14_2.txt
!!
2615 bytes copied in 0.556 secs
```

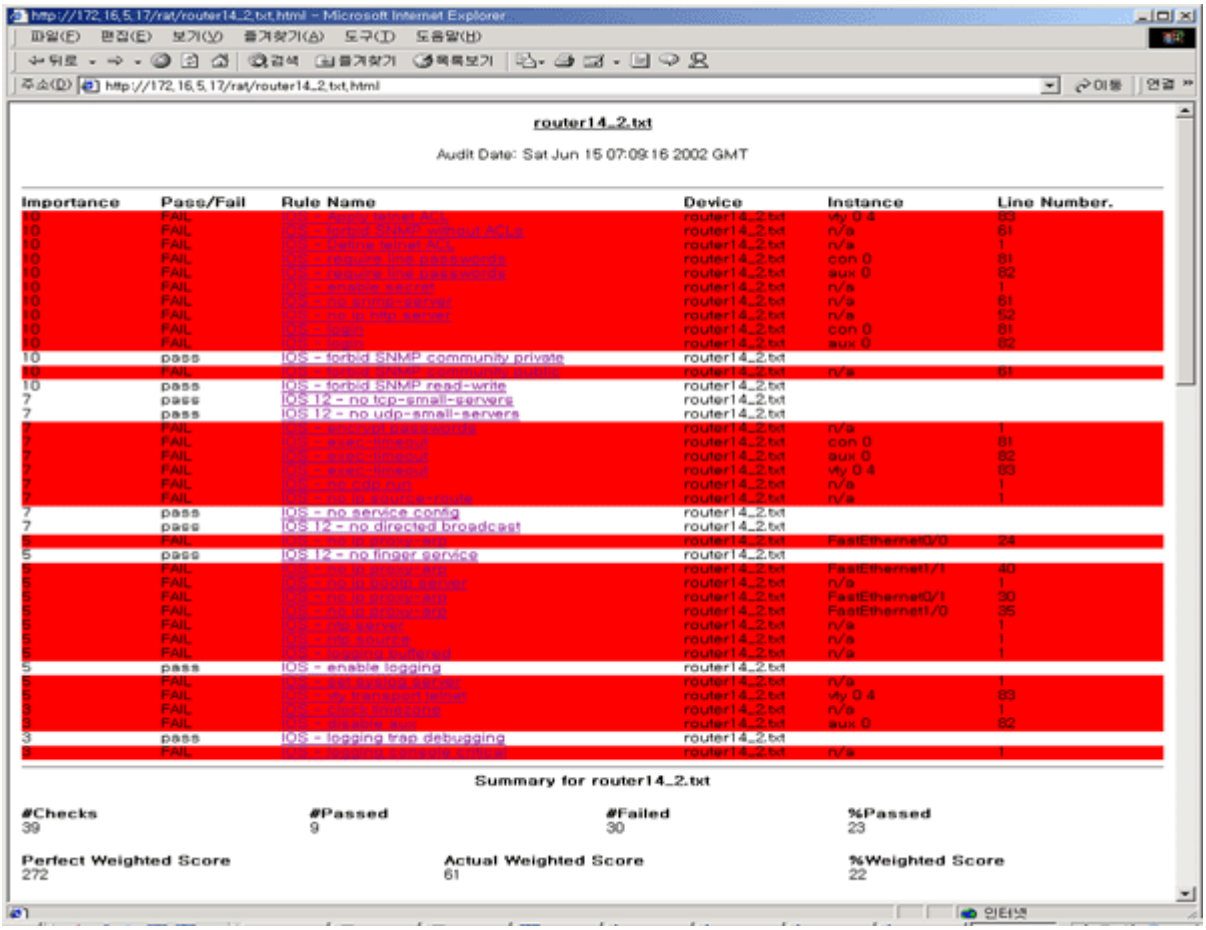
③ 환경설정파일을 결과파일이 위치할 디렉토리로 옮겨서 아래와 같은 명령을 실행한다.
문법 : rat [OPTIONS] config [config ...]

```
# rat router14_2.txt
auditing router14_2.txt...done.
ncat_report: writing router14_2.txt.ncat_fix.txt.
ncat_report: writing router14_2.txt.ncat_report.txt.
ncat_report: writing router14_2.txt.html.
ncat_report: writing rules.html (cisco-ios-benchmark.html).
ncat_report: writing all.ncat_fix.txt.
ncat_report: writing all.ncat_report.txt.
ncat_report: writing all.html.
```

이제 현재 디렉토리에 점검결과 파일들이 생성된 것을 확인할 수 있을 것이다.

4. 점검결과 및 조치

결과 파일은 웹브라우저를 통해서 확인할 수 있다.
다음은 그림은 점검결과를 나타내고 있는데 각 점검항목에서 "Pass(안전)"한 항목은 흰색으로, "Fail(취약)"한 항목은 붉은색으로 표시가 되어 진다(그림 1).



router14_2.txt

Audit Date: Sat Jun 15 07:09:16 2002 GMT

Importance	Pass/Fail	Rule Name	Device	Instance	Line Number.
10	FAIL	IOS - apply telnet ACL	router14_2.txt	vty 0 4	83
10	FAIL	IOS - forbid SNMP without ACLs	router14_2.txt	n/a	61
10	FAIL	IOS - define telnet ACL	router14_2.txt	n/a	1
10	FAIL	IOS - require line passwords	router14_2.txt	con 0	81
10	FAIL	IOS - require line passwords	router14_2.txt	aux 0	82
10	FAIL	IOS - no enable server	router14_2.txt	n/a	1
10	FAIL	IOS - no small-servers	router14_2.txt	n/a	61
10	FAIL	IOS - no ip http server	router14_2.txt	n/a	52
10	FAIL	IOS - telnet	router14_2.txt	con 0	81
10	FAIL	IOS - telnet	router14_2.txt	aux 0	82
10	pass	IOS - forbid SNMP community private	router14_2.txt		
10	FAIL	IOS - forbid SNMP public	router14_2.txt	n/a	61
10	pass	IOS - forbid SNMP read-write	router14_2.txt		
7	pass	IOS 12 - no tcp-small-servers	router14_2.txt		
7	pass	IOS 12 - no udp-small-servers	router14_2.txt		
7	FAIL	IOS - service password-encrypt	router14_2.txt	n/a	1
7	FAIL	IOS - exec-timeout	router14_2.txt	con 0	81
7	FAIL	IOS - exec-timeout	router14_2.txt	aux 0	82
7	FAIL	IOS - exec-timeout	router14_2.txt	vty 0 4	83
7	FAIL	IOS - no cdp run	router14_2.txt	n/a	1
7	FAIL	IOS - service password-encrypt	router14_2.txt	n/a	1
7	pass	IOS - no service config	router14_2.txt		
7	pass	IOS 12 - no directed-broadcast	router14_2.txt		
5	FAIL	IOS - no finger service	router14_2.txt	FastEthernet0/0	24
5	pass	IOS 12 - no finger service	router14_2.txt		
5	FAIL	IOS - no ip-source-arp	router14_2.txt	FastEthernet1/1	40
5	FAIL	IOS - no ip boot server	router14_2.txt	n/a	1
5	FAIL	IOS - no ip-source-arp	router14_2.txt	FastEthernet0/1	30
5	FAIL	IOS - no ip-source-arp	router14_2.txt	FastEthernet1/0	35
5	FAIL	IOS - no ip server	router14_2.txt	n/a	1
5	FAIL	IOS - no ip source-rt	router14_2.txt	n/a	1
5	FAIL	IOS - no ip bootp server	router14_2.txt	n/a	1
5	pass	IOS - enable logging	router14_2.txt		
5	FAIL	IOS - set logging server	router14_2.txt	n/a	1
5	FAIL	IOS - vty transport telnet	router14_2.txt	vty 0 4	83
3	FAIL	IOS - logging timestamp	router14_2.txt	n/a	1
3	FAIL	IOS - logging buffered	router14_2.txt	aux 0	82
3	pass	IOS - logging trap debugging	router14_2.txt		
3	FAIL	IOS - logging console critical	router14_2.txt	n/a	1

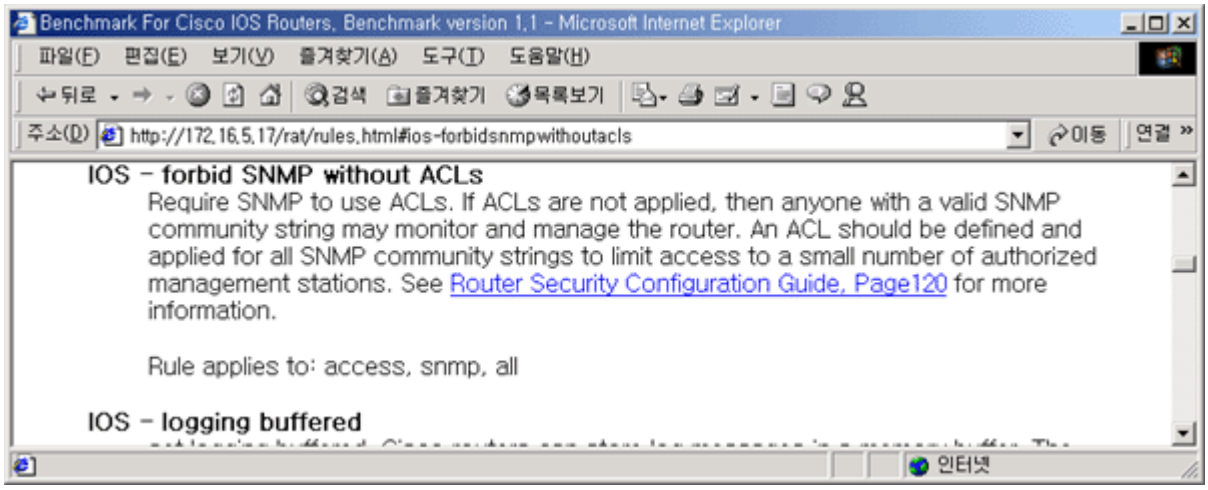
Summary for router14_2.txt

#Checks	#Passed	#Failed	%Passed
39	9	30	23

Perfect Weighted Score	Actual Weighted Score	%Weighted Score
272	61	22

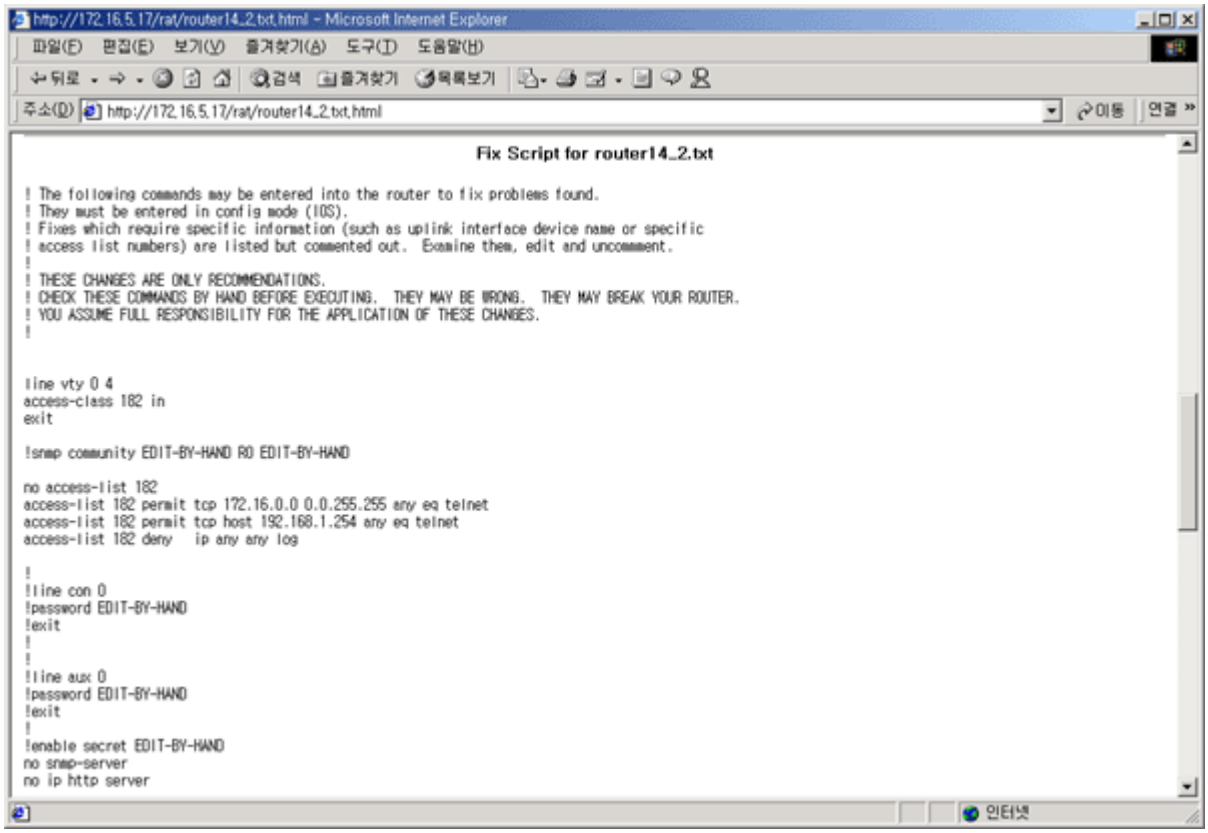
(그림 1) RAT 점검 결과

또한, 각 점검항목에 대해서는 "Rule Name"이 hyper-text 형태로 취약점에 대한 설명 및 조치방법을 소개해 주고 있다(그림 2).



(그림 2) 각 점검항목에 대한 설명

그리고, RAT에서 취약한 부분에 대한 설정을 바꿔서 권장하는 설정파일의 예도 제시하고 있어 라우터 관리자들이 쉽게 따라서 설정할 수 있도록 하고 있다(그림 3).



(그림 3) RAT에 의해 권장된 환경설정

RAT에서 권장하는 환경설정을 단순히 Copy & Paste 하여 각자의 라우터 Config 모드에서 적용시킬 수 있을 것이다.

5. 결론

지금까지 라우터 보안점검 도구인 RAT의 기능, 설치과정, 점검결과 분석 등에 대해서 살펴보았다.

최근 각종 시스템의 보안버그나 설정오류를 이용한 인터넷 웜에 의한 피해규모가 증가하고 있는 가운데, 라우터를 공격대상으로 하는 새로운 웜의 등장가능성에 대해서도 조심스럽게 이야기되고 있다.

라우터의 오동작은 곧 네트워크의 단절로 이어질 수 있는 크나큰 문제가 아닐 수 없다. 라우터의 중요성과 보안위협요소들은 크게 증가하고 있지만 일상적인 보안점검에서도 제외되는 등 관리의 사각지대였던 것이 사실이다.

보안점검 도구들도 Unix, Linux, Windows 등 각종 운영체제에 대한 보안취약점 점검도구들은 상용/공개도구들이 쏟아져 나왔었지만 라우터에 대한 점검도구를 찾아보기는 쉽지 않았다.

이러한 가운데 RAT는 라우터의 다양한 보안설정을 자동으로 점검할 수 있는 아주 훌륭한 도구라고 생각되어 진다.
RAT는 다수의 라우터를 대상으로 각종 설정값을 자동으로 점검하고 신속하게 설정값들을 바꿀 수 있도록 도와주며, 라우터 설정값의 변화를 주기적으로 감시하는데도 도움을 줄 수 있다.