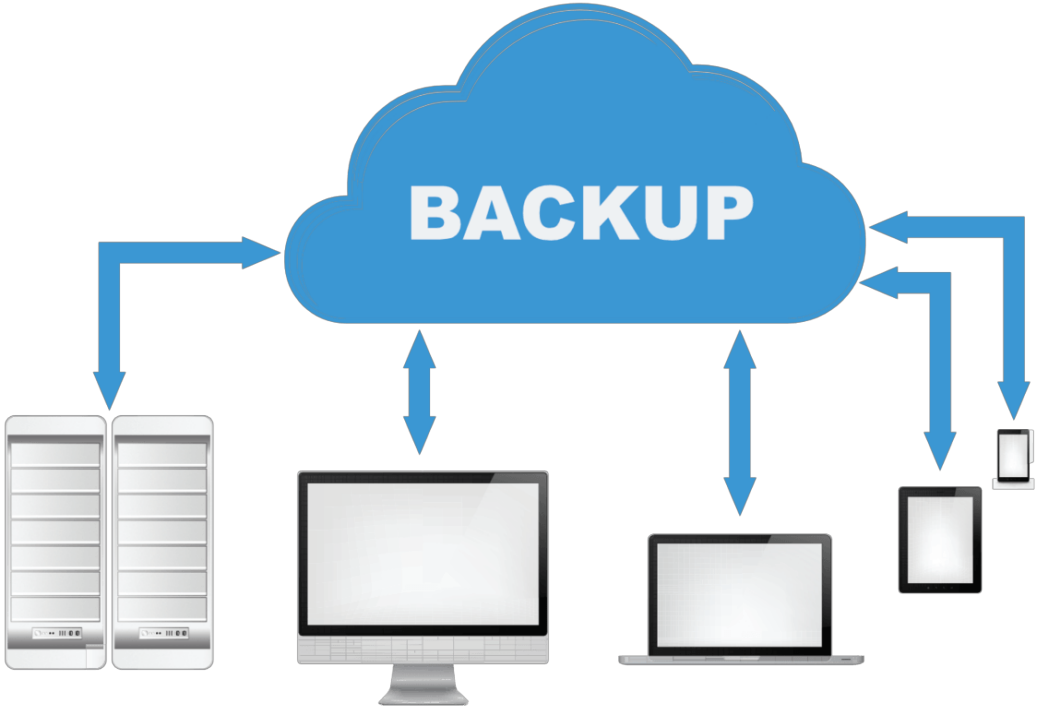
Back Up

Kris

What is Backup?

- ~~Something you do if you have a working brain.~~
- Saving data somewhere else
- Preparing for Emergency



Why Back Up?

안원용 ▶ 생활코딩
2월 19일 오후 1:45 ·

rm -rf /* 전설로만 내려오는 명령어를, 어제 밤 새벽 1시에 서비스중인 CENTOS 서버에서 폴더 하나 지워야지 하면서 입력했습니다.

이기윤 ▶ 생활코딩
4월 10일 오후 4:26 ·

후배가 장장 2주동안 열심히 만든 라즈베리파이의 서버를 모르고 rm -rf/* 입력했는데 어찌 죽여야 잘죽였다고 소문날까요.

유용민 ▶ 생활코딩
2017년 5월 8일 오후 10:59 ·

코딩동아리에서 활동하는 고2 학생입니다. 학교컴터는 재부팅하면 포맷되는지라 USB에 우분투를 설치해서 들고다니며 썼습니다.. 라이브 말고 설치요.. 오늘 우분투에서 프로젝트 하나 날린다고.. 무슨생각인지.. rm -rf /usr/local/projects/Project1/* 을 해야하는데.... rm -rf /* 을 쳐버렸습니다.. 무려 root로 로그인된... 더 보기

8 댓글 21개

Why Back Up?

- 하드웨어에 물리적인 문제가 생겼거나
- 실수로 파일/폴더를 삭제했거나
- 컴퓨터를 잃어버렸거나
- 컴퓨터를 도둑맞았거나
- 랜섬웨어같은 바이러스에 걸렸거나

조금은 다른 사례지만,

- 게임을 악용하는 사례가 생겨 어느 시점으로 롤백을 해야할 때도 백업이 필요합니다.

Causes of Loss of Data

- 사람의 실수
- 관리자나 해커에 의한 고의적인 파괴
- 악성코드(랜섬웨어)
- 저장장치의 물리적인 결함
- 전원 이상(전기적인 문제)
- 과열
- OS 또는 프로그램에 의한 오류
- 도난 및 분실

What Do I Save?

- Anything!
- Server Data, Client Data, Pictures, Documents, Websites, VPN Server Information, Etc…

Type of Back Ups

- Full backup
- Incremental backup
- Differential backup
- Mirror backup

Differences

	백업되는 데이터	백업 시간	복구 시간	백업 공간
Full backup	전부	매우느림	빠름	높음
Incremental backup	마지막 Full Backup 이후 변경점	빠름	보통	아주 낮음
Differential backup	마지막 Backup 이후 변경점	보통	빠름	보통
Mirror backup	마지막 Full Backup 이후 변경점	아주 빠름	아주 빠름	아주 높음

Full Backup

- 백업할 때마다 모든 내용을 백업한다.
- 백업하는 내용을 모두 압축하여 하나의 파일로 저장한다.
- 시간 소모는 많으나 복구가 간단하며 저장 공간을 많이 차지한다.
- 해킹 당할 경우 모든 내용을 내어주기 때문에 암호화가 필요하다.
- 다른 종류의 백업이 full backup에 의존하므로 주기적으로 해 주어야 한다

Incremental Backup

- 마지막 백업 이후로 바뀐 내용을 백업한다.
- 적은 내용을 백업하므로 백업 속도는 빠르다
- 하지만 복구 시에는 여러 파일을 각각 백업해야 하므로 시간이 오래 걸린다.
- MacOS의 Time Machine이 이 방식을 채용한다.

Differential Backup

- 마지막 full backup 이후로 바뀐 내용을 백업한다.
- 백업 속도는 느리나 복구 속도는 빠르다.
- 차지하는 용량은 중간 정도이다.

Mirror Backup

- 마지막 full backup 이후로 바뀐 내용을 저장한다.
- 파일 내용을 압축 없이 그대로 저장하여 저장 공간을 많이 차지하며 보안에 취약하다.
- 백업 속도와 복구 속도가 가장 빠르다.

Back up file with .tar

- $ sudo mkdir backups
- $ cd backups
- $ sudo tar -cvpf /backups/full-backup.tar —directory=/ —exclude=proc —exclude=sys —exclude=dev/pts —exclude=backups .

-c: tar로 묶는다
-v: 압축 과정을 터미널에 출력한다
-p: 파일 권한을 함께 저장한다
-f: 파일 이름을 지정한다

Back up files with .gz, .bz2

- $ sudo mkdir backups
- $ cd backups
- $ sudo tar -zcvpf /backups/full-backup.tar.gz (or .bz2) —directory=/ —exclude=proc —exclude=sys —exclude=dev/pts —exclude=backups .

bzip2가 gzip에 비해 압축 및 풀기 속도는 낮으나 더 용량이 작아짐

-z 옵션을 줘야 zip으로 압축할 수 있다!

Back up in remote server

- 보통은 서버에 문제가 생길 때도 많기 때문에 같은 서버에 백업을 해놓는 것은 좋지 않음
  (ex. rm –rf /*)
- 다른 백업 서버를 만들어서 운용하는 것이 좋다.

Rsync

- rsync는 원격 파일 복사 프로그램
- 소스와 대상 파일을 비교하여 변경된 내용만 전송
- 그래서! 자료 전송량이 적고 빠르다.
- rsync [option] [source(file to send)] [destination]

Rsync - options

- -v or —verbose 자세하게 출력
- -q or —quiet 어떤 메시지도 출력하지 않음 (에러 포함)
- -a or —archive 아카이빙
  (위치, 권한, 소유주 포함하여 가져옴)
- -r or —recursive 하위 구조의 디렉토리도 recursive하게
- -z or —compress 압축해서 전송

Rsync - example

```
sparcs@46a0fba8a998:~/newbie_project$ sudo rsync -v index.js ubuntu@13.125.175.179:/home/ubuntu
ubuntu@13.125.175.179's password:
Permission denied, please try again.
ubuntu@13.125.175.179's password:
index.js

sent 4,234 bytes  received 35 bytes  656.77 bytes/sec
total size is 4,148  speedup is 0.97
```

```
ubuntu@ip-172-31-36-160:~$ ls
index.js  parang
ubuntu@ip-172-31-36-160:~$ cat index.js
const express = require('express');
const app = express();
const bodyParser = require('body-parser');
const User = require('./model/user.js');
const History = require('./model/history.js');
const cors = require('cors');
var request = require('request');
app.use(cors());
app.use(express.static('static'));
```

Rsync – ssh with rsa

$ sudo apt-get install ssh rsync

$ ssh-keygen -t rsa

(RSA 타입으로 public key와 private key를 발급)

$ scp .ssh/id_rsa.pub ubuntu@13.125.175.179:~/.ssh/

authorized_keys

$ rsync -avz —progress -e ssh /backups

ubuntu@13.125.175.179:backups

Rsnapshot

- rsync 기반의 파일 시스템 백업 유틸리티
- Incremental backup을 사용하여 용량을 적게 차지한다.

Rsnapshot

$ sudo apt-get install rsnapshot

$ sudo vi /etc/rsnapshot.conf

Config options:

- snapshot_root: 백업이 저장될 폴더의 절대 경로
- cmd_ssh 라인의 주석을 해제하면 원격 백업이 가능
- retain [name] [n]: 해당 이름의 백업을 n개까지 유지
- verbose, loglevel: 터미널 출력 메시지와 로그를 어느 정도 상세하게 기록할 것인지

Rsnapshot - Restore

$ mkdir here

$ sudo cp -r /var/cache/rsnapshot/alpha.0 here

Rsnapshot - example

```
ubuntu@ip-172-31-36-160:~$ sudo rsnapshot -v alpha
echo 21707 > /var/run/rsnapshot.pid
mkdir -m 0755 -p /var/cache/rsnapshot/alpha.0/
/usr/bin/rsync -a --delete --numeric-ids --relative --delete-excluded \
    /home/ /var/cache/rsnapshot/alpha.0/localhost/
/usr/bin/rsync -a --delete --numeric-ids --relative --delete-excluded /etc/ \
    /var/cache/rsnapshot/alpha.0/localhost/
/usr/bin/rsync -a --delete --numeric-ids --relative --delete-excluded \
    /usr/local/ /var/cache/rsnapshot/alpha.0/localhost/
/usr/bin/rsync -a --delete --numeric-ids --relative --delete-excluded \
    /test/ /var/cache/rsnapshot/alpha.0/localhost/
touch /var/cache/rsnapshot/alpha.0/
rm -f /var/run/rsnapshot.pid
```

Backups in General

보통은 매번 백업해주기 힘드니까 쉘스크립트를 쓴다.

정기적으로 하려면?

미리 짜여진 쉘스크립트를 정기적으로 실행해주면 끝.

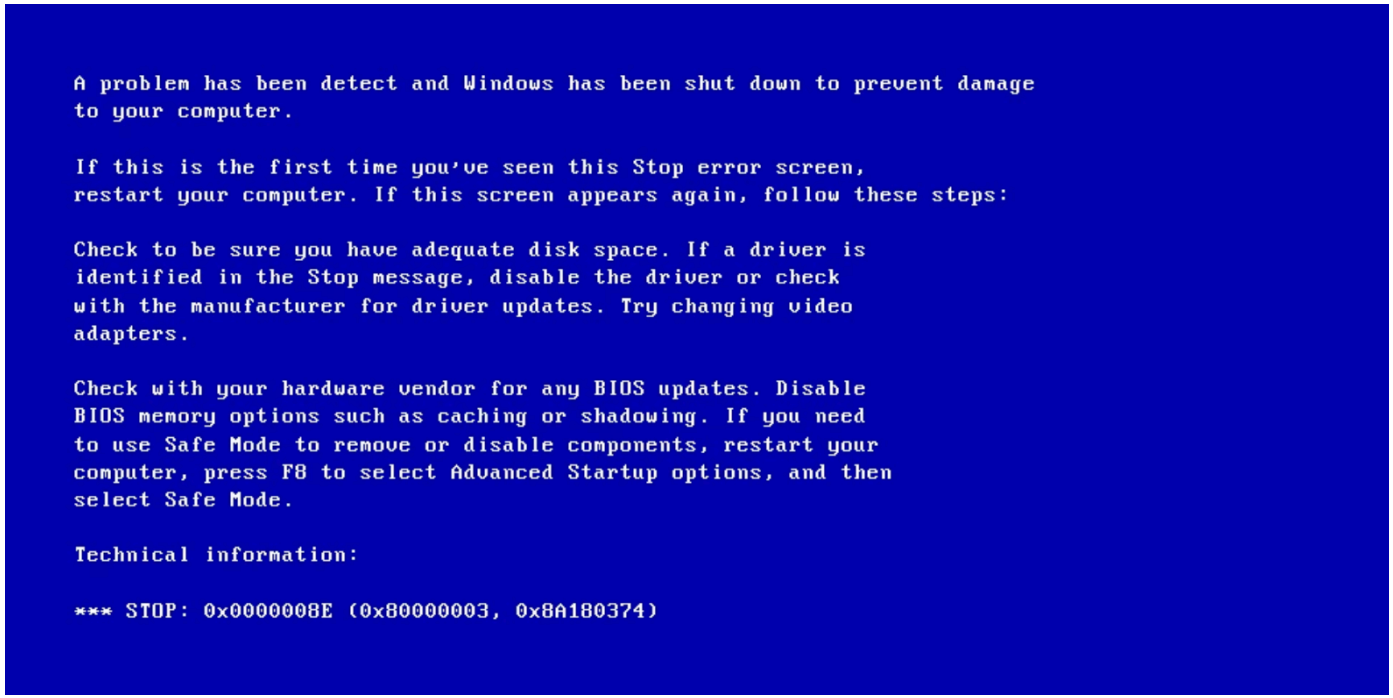
또는, 앞서 배운 크론 세미나에서 크론을 이용하는 방법도 존재한다. ~~솔직히 크론 쓸 줄 알면 그냥 크론으로 하자...~~

Other Backup Tips

- 오래된 백업 파일은 용량 절감을 위해 삭제하거나 시간 간격을 넓게 하여 저장 공간을 효율적으로 사용한다.
- 백업 디스크를 읽기 전용으로 설정하거나, 하드웨어로 쓰기 및 변조 방지 옵션을 걸어 랜섬웨어 등의 공격에 대비한다.
- 백업 디스크를 만든 후 백업 전 mount, 백업 후 unmount 하는 방식으로 rm -rf 공격을 막을 수 있다.

Kernel Panic

- OS가 복구하기 어려운 치명적인 오류를 발견했을 때 발생
- 장치 오작동, 메모리 문제 등 다양한 Cause
- 블루스크린이 가장 대표적인 예

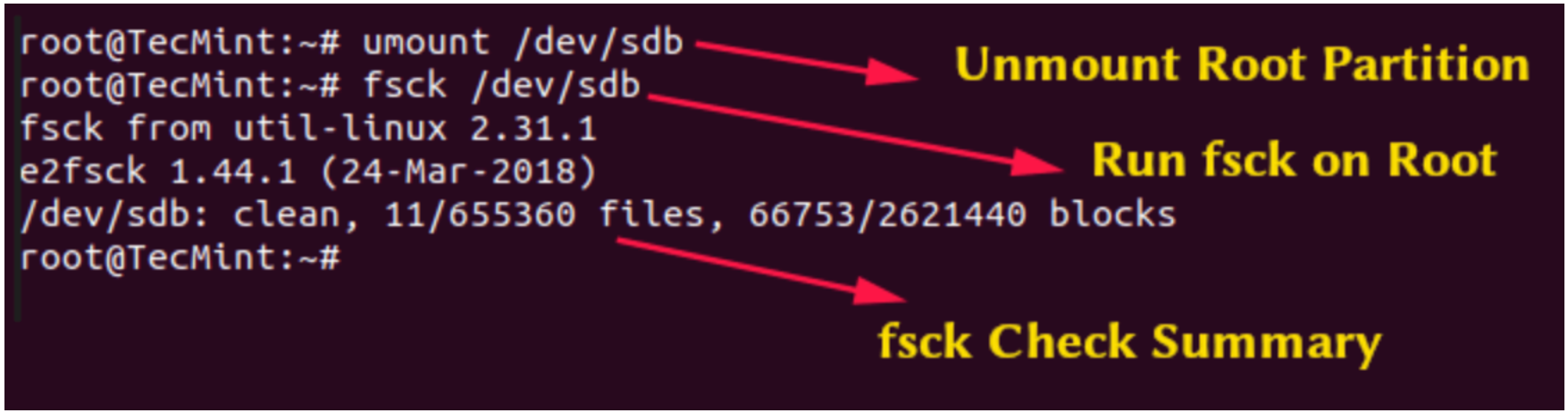


Hardware Failures

- Broken file system – Most common
  - e2fsck 이라는 명령어로 복구 및 점검
- /etc/fstab 에서 잘못 설정된 경우
  - Linux Secure로 부팅 후 수정, 그리고 재부팅
- 이 외에 물리적인 문제점들
  - 단선, 랜선 문제, 먼지, 접촉불량, 연결 등 아주 많다.
  - 단순 고장의 경우 선 뺐다 꼽기, 재부팅, 먼지털기 등 기본적인 조치를 취해 준 후, 죽었다 싶으면 새로 사버리자
  - 삐삐삐 경고음 소리는 대체로 냉각/전원 문제라고 한다.

e2fsck

- 리눅스 파일 시스템을 점검 및 복구할 수 있는 명령어
- fsck의 새로운 버전
- 기본적으로 점검하는 목록
  - inodes, blocks, sizes, 디렉토리 구조, 디렉토리 연결성, 파일링크, 전체 파일 개수, 전체 블록 중 사용중인 블록
- $ e2fsck [options] [검사할 디바이스 이름]
- 해당 Filesystem을 Unmount해야 사용가능하다



정전

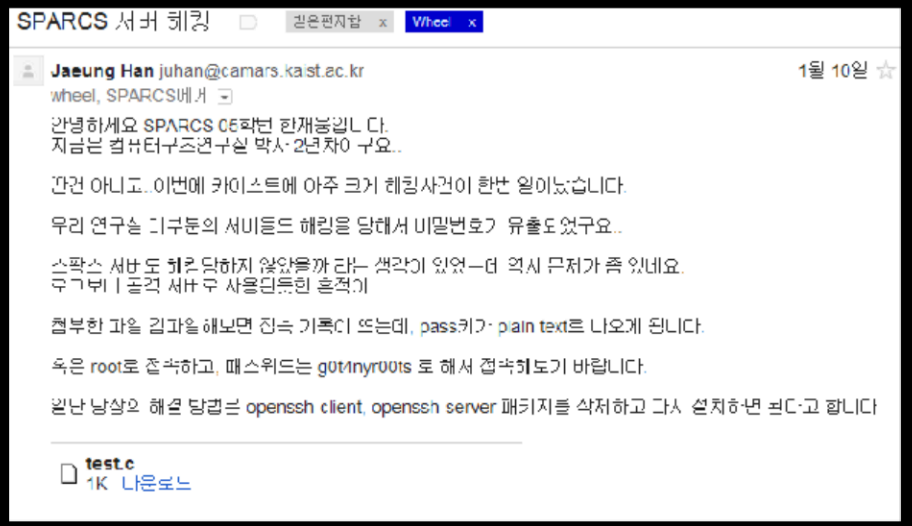
- 정전 발생시 당황하지 말자.
  - ~~(사실 정전나도 서버부터 걱정하는 사람이 몇이나 될까)~~
  - AWS로 완전이전 된다면 신경쓸 필요가 없어진다
- 서버실에는 UPS(무정전전원공급장치)가 있어서 어느정도의 시간의 서버들이 멈추지 않고 작동할수 있다.
- 서버를 종료하기 전에 SPARCS로 서비스 중단 메일공고를 하고, 서버들을 차례대로 종료하도록 하자.

서버실

- 서버실의 온도는 겨울철 26도, 여름철 33도에서 37도에 이르기까지 좀 높은 편이다. 서버실에 온도계가 설치되어 있으니 자주 확인할 수 있다.
- 서버실 온도가 높다면 일단 서버실에 들어가, 에어컨의 작동여부를 확인해보고, 에어컨이 고장난 경우, 리모컨으로 전원을 다시 켜보고 온도를 최저로 낮춰보자. 가끔 멀티탭에 문제가 있는 경우도 있다.
- 에어컨에 문제가 있다면 시설팀에 연락하고, 수리를 요청하자.
- 서버실의 문을 열고, 선풍기등을 이용해 열을 빼내도록 하자. (단 보안에 신경써야 한다)
- 비상시, 중요하지 않은 서버들 (다래, ftp2, mir 등)을 종료하여 최대한 발열량을 줄이자.

2012 KAIST/SPARCS 해킹 사건

- 2012년 1월 10일.
- KAIST 연구실들이 해킹당함. 스팍스도?!
- /var/run/sshd.sync/에 사용자 비밀번호가 plain text로 저장.
- root 아이디에 g0t4nyr00ts 를 치면 접속이 된다.


SPARCS 서버 해킹

받은편지함 x Wheel x

Jaeung Han juhan@camars.kaist.ac.kr 1월 10일

wheel, SPARCS에서

안녕하세요 SPARCS CS학번 한재웅입니다.
지금은 컴퓨터구조연구실 박사 2년차이구요..

간간 아니고..이번에 카이스트에 아주 크게 해킹사건이 한번 일어났습니다.

우리 연구실 그루둔의 서버들도 해킹을 당해서 비밀번호가 유출되었구요..

스팍스 서버도 해킹당하지 않았을까 라는 생각이 있었는데 역시 문제가 좀 있네요.
우리뷔 | 공격 서버 두 사용된듯한 흔적이

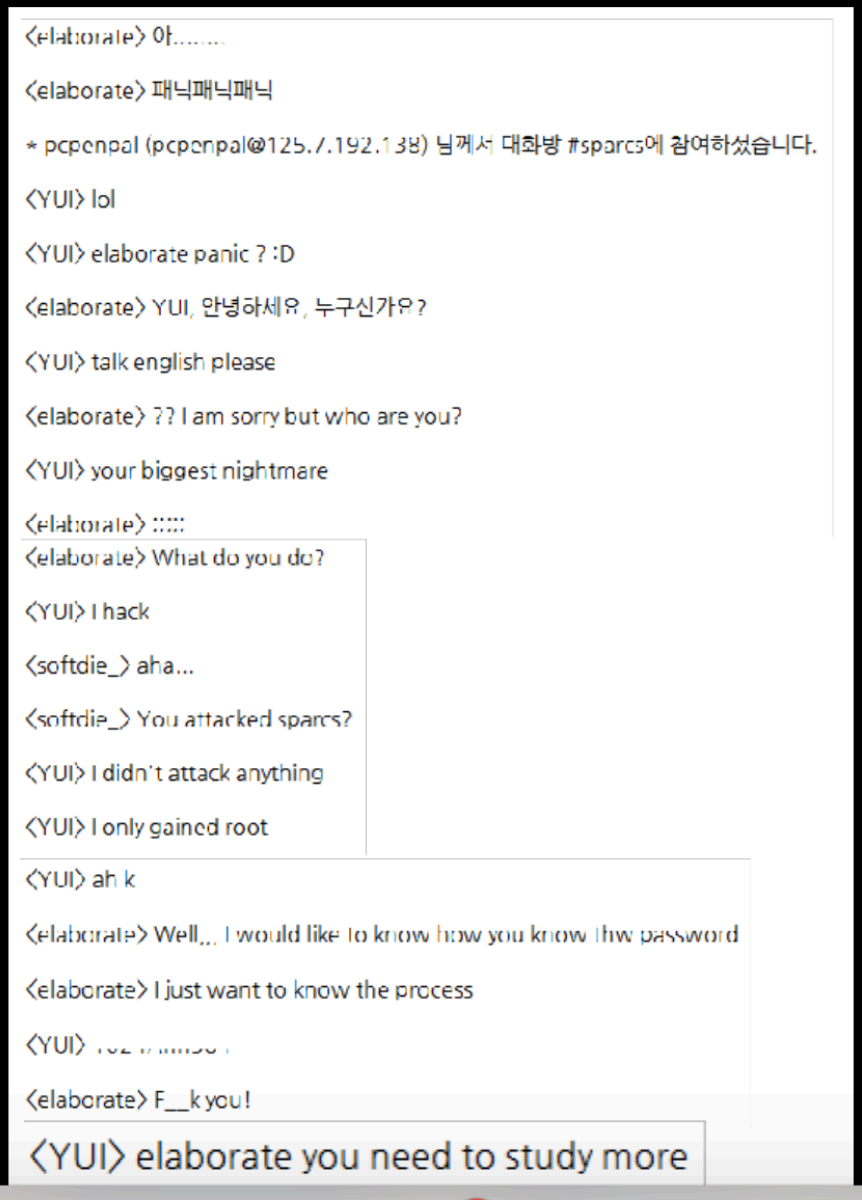
첨부한 파일 실과을 해보면 진속 기록이 뜨는데, passwd가 plain text로 나오게 됩니다.

속은 root로 접속하고, 패스워드는 g0t4nyr00ts 로 해서 접속하도기 바랍니다.

일단 당장은 해결 방법은 openssh client, openssh server 패키지를 삭제하고 다시 설치하면 된다고 합니다

test.c
1K 다운로드

2012 KAIST/SPARCS 해킹 사건



<YUI>란 아이디의 해커가
스팍스 IRC에 접속

2012 KAIST/SPARCS 해킹 사건

- 해킹 루트 분석:

  zeroboard4 취약점으로 root획득

-> 획득한 서버에 백도어 설치

-> 획득한 서버의 sshd를 변조

-> ssh 사용자의 id와 pw를 획득

-> 다른 서버에 같은 작업 반복

Thank you!