Backdooring Git

John Menerick – August 2015
Legal Disclaimer
Thank you for coming
What we are covering
What we are not covering
<table>
<thead>
<tr>
<th>Software</th>
<th>Maintainer</th>
<th>Development status</th>
<th>Reporting model</th>
<th>Dissemination model</th>
<th>License</th>
<th>Platforms supported</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcroRae 25</td>
<td>Micro Pace International</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, macOS, Linux, FreeBSD, Modern OS</td>
<td>Non-free Quoted on an individual basis</td>
</tr>
<tr>
<td>Omni Planner</td>
<td>Central Ltd</td>
<td>limited development, latest release August 4, 2013</td>
<td>Distributed and client-server</td>
<td>Merge</td>
<td>GNU GPL</td>
<td>Linux, FreeBSD, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Bitkeeper</td>
<td>BitKeeper Inc.</td>
<td>Active</td>
<td>Distributed</td>
<td>Merge</td>
<td>Proprietary</td>
<td>Linux, Windows, OS X</td>
<td>Non-free Quoted on an individual basis</td>
</tr>
<tr>
<td>VS Software</td>
<td>CA Technologies</td>
<td>CA Software Change Manager</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or Lock</td>
<td>Proprietary</td>
<td>Unix, Linux, Windows, VMware</td>
</tr>
<tr>
<td>CodeCheck</td>
<td>IBM Rational</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>GNU GPL</td>
<td>Linux, Windows, OS X</td>
<td>Non-free $4800 per floating license (eastern time zone) per person per year, can be complicated multiply</td>
</tr>
<tr>
<td>Core i5 vs-2600k</td>
<td>Redgate Software</td>
<td>Active</td>
<td>Proprietary</td>
<td>Windows</td>
<td>Proprietary</td>
<td>Windows</td>
<td>Non-free $150 per seat</td>
</tr>
<tr>
<td>DiffY</td>
<td>Rove Cohen</td>
<td>official site offers latest release July 15, 2007</td>
<td>Distributed</td>
<td>Merge</td>
<td>proprietary codebase merge</td>
<td>BSD</td>
<td>Linux, Windows, OS X</td>
</tr>
<tr>
<td>CVS</td>
<td>The CVS Team</td>
<td>maintained but new features not added</td>
<td>Distributed</td>
<td>Merge</td>
<td>GNU GPL</td>
<td>Linux, Windows, OS X</td>
<td>Free with (CVS distribution key) or by written or email commercial license for latest version of CVS Suite or Change Management Server</td>
</tr>
<tr>
<td>CVSI</td>
<td>Mark H. Yacson Software</td>
<td>maintained and new features under development</td>
<td>Distributed</td>
<td>Merge or lock</td>
<td>GNU GPL, BSD, or proprietary</td>
<td>Unix, Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>CVS</td>
<td>The The CVS team</td>
<td>Distributed</td>
<td>Merge</td>
<td></td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
<td></td>
</tr>
<tr>
<td>Dimension CM</td>
<td>Siemens Software</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Linux, Solaris, AIX, HP-UX, Solaris, and Unix</td>
<td>Free</td>
</tr>
<tr>
<td>Flex</td>
<td>John Renshaw</td>
<td>Active</td>
<td>Distributed</td>
<td>Merge</td>
<td>BSD</td>
<td>POSIX, Windows, OS X, Solaris</td>
<td>Free</td>
</tr>
<tr>
<td>GNU-rsync</td>
<td>Andy Tan</td>
<td>unmirrored</td>
<td>Distributed</td>
<td>Merge</td>
<td>GNU GPL</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Hg</td>
<td>Mercurial</td>
<td>active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, OS X</td>
<td>Non-free Commercial</td>
</tr>
<tr>
<td>IDES</td>
<td>Magista Corp, Inc.</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge</td>
<td>Proprietary</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Illinois Synthesizer</td>
<td>Astelco respectively</td>
<td>latest release May 21, 2006</td>
<td>Client-server extended to 'yes'</td>
<td>Merge</td>
<td>GNU GPL [1]</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Mercurel</td>
<td>Hall McDowell</td>
<td>Active</td>
<td>Distributed</td>
<td></td>
<td>GNU GPL, BSD, Open Source</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Minutiae</td>
<td>Nathaniel Olsen, Clayton Woolard</td>
<td>Active</td>
<td>Distributed</td>
<td></td>
<td>GNU GPL, BSD, Open Source</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>PerlPack</td>
<td>Phenom Software Inc.</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, OS X</td>
<td>Free</td>
</tr>
<tr>
<td>Poker SCM</td>
<td>Carbon Software</td>
<td>Active</td>
<td>Distributed</td>
<td>Merge</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows X</td>
<td>Free</td>
</tr>
<tr>
<td>PWS</td>
<td>Cisco Systems</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows X</td>
<td>Free</td>
</tr>
<tr>
<td>Software Maintenance</td>
<td>National Science Foundation</td>
<td>Active</td>
<td>Distributed</td>
<td>Merge</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows X</td>
<td>Free</td>
</tr>
<tr>
<td>Rational Team Credit</td>
<td>IBM Rational</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows, AIX, HP-UX, Solaris, and OS X</td>
<td>Free for up to 12 users, else starting at $595 per seat or $2,400 per 20 development per year [2]</td>
</tr>
<tr>
<td>Rational Product</td>
<td>IBM Rational</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows, AIX, HP-UX, Solaris, and OS X</td>
<td>Free for up to 12 users, else starting at $595 per seat or $2,400 per 20 development per year [2]</td>
</tr>
<tr>
<td>Reason Control Solution</td>
<td>IBM Rational</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, UNIX, Windows, AIX, HP-UX, Solaris, and OS X</td>
<td>Free for up to 12 users, else starting at $595 per seat or $2,400 per 20 development per year [2]</td>
</tr>
<tr>
<td>Source Code</td>
<td>Jugi Solutions</td>
<td>Active</td>
<td>Merged</td>
<td></td>
<td>GNU GPL</td>
<td>Proprietary</td>
<td>Linux, Windows, Unix, OS X</td>
</tr>
<tr>
<td>Source Code Management</td>
<td>Dynamic Corporation</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, Unix, OS X</td>
<td>Free for up to 12 users, else starting at $595 per seat or $2,400 per 20 development per year [2]</td>
</tr>
<tr>
<td>BarTaps</td>
<td>Bernard Schöler (Bernhardt)</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows and Cross-platform via web service interface</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Subversion (SVN)</td>
<td>Apache Software Foundation</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, Unix, OS X, FreeBSD</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Sunsmith SCM</td>
<td>Sunshine Software</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, Unix, OS X</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>svn</td>
<td>Sunsmith SCM</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Linux, Windows, Unix, OS X</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Team Foundation</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Team Foundation</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Source Royalty</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Source Royalty</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Source Royalty</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Source Royalty</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
<tr>
<td>Source Royalty</td>
<td>Microsoft</td>
<td>Active</td>
<td>Client-server</td>
<td>Merge or lock</td>
<td>Proprietary</td>
<td>Windows, Cross-platform via Visual Studio Onsite</td>
<td>Free for up to 5 users in the Visual Studio Onsite or for open source projects through codebase.com, else non-free, licensed through MSDN subscription or direct buy.</td>
</tr>
</tbody>
</table>
"Software is like sex; it's better when it's free."

- Bill Gates
Setting the Stage
Good luck!

NO NEED
FOR SOURCE
CONTROL.
Revision control vs. Source Control

Source control == source code change management
Wrong Tool for the Job
Right Tool for the Job
Distributed vs. Centralized

Traditional

Distributed
Helfe!

Help me Jesus, help me Jewish God, help me Allah
HELP ME TOM CRUISE
Trends

Beta: Measuring search interest in topics is a beta feature which quickly provides accurate measurements of overall search interest. To measure search interest for a specific query, select the "search term" option.

Interest over time

News headlines

Forecast
1 While it works, angel sings and light shines from above - “Global information tracker”
2 When it dies, fire erupts from under your feet - “Goddamn idiot truckload of sh*t”
Hitler Uses Git
Rings of Trust
If you have ever done any security work - and it did not involve the concept of “network of trust” - it wasn’t security work, it was - <insert word my mother would not approve me stating>. I don’t know what you were doing. But trust me, it’s the only way you can do security. it’s the only way you can do development.

Linus Torvalds
Typical Trust Relationships

- Dictator
- Lieutenant
- Blessed Repository
- Developers
Since you do not want everybody to write to the central repository because most people are morons, you create this class of people who are ostensibly not morons. And most of the time what happens is that you make that class too small, because it is really hard to know if a person is smart or not, and even if you make it too small, you will have problems. So this whole commit access issue, which some companies are able to ignore by just giving everybody commit access, is a huge psychological barrier and causes endless hours of politics in most open source projects.
Empirical Study
Outstanding vs Fixed defects over period of time

- Feb 22, 2010
Git

High impact Outstanding Defect per Category

- Resource leaks
- Memory - illegal accesses
- Memory - corruptions

Outstanding defects

Medium impact Outstanding Defect per Category

- Null pointer dereferences
- Error handling issues
- Insecure data handling
- Integer handling issues
- Control flow issues
- Incorrect expression
- Concurrent data access violations

Outstanding defects
Not Scientific CVE Search

Found: 2129 Secunia Security Advisories, displaying 1-25

Sort by: Match, Title, Date

Title
- GIT Case-insensitive Filesystem Data Manipulation Vulnerability
- GIT “git-map-send” SSL Certificate Verification Security Issue
- GIT “gitweb” Cross-Site Scripting Vulnerabilities
- GIT “is_git_directory()” Buffer Overflow Vulnerability
- Git git-daemon Parameter Parsing Infinite Loop Denial of Service
- GIT “gitweb” Command Injection Vulnerabilities
- GIT “gitweb” Privilege Escalation Security Issue
- GIT Pathname Processing Multiple Buffer Overflows
- GIT “git-checkout-index” Symbolic Link Handling Buffer Overflow

Found: 7 Secunia Security Advisories, displaying 1-7

Sort by: Match, Title, Date

Title
- Perforce Web Client (P4Web) Multiple Cross-Site Scripting Vulnerabilities
- Perforce Server Multiple Vulnerabilities
- Perforce P4FTP FTP Plugin Denial of Service
- Perforce P4Web Client Two Vulnerabilities
- Perforce Server Multiple Vulnerabilities
- Perforce Server Denial of Service Vulnerabilities
- Perforce Web Client HTTP Request Processing Denial of Service

Found: 438 Secunia Security Advisories, displaying 1-25

Sort by: Match, Title, Date

Title
- WebSVN Symlink Arbitrary File Download Vulnerability
- Perl SVN::Look Module Command Injection Vulnerability
- SVNManager Multiple SQL Injection Vulnerabilities
- WebSVN "path" Cross-Site Scripting Vulnerability
- WebSVN Shell Command Injection Vulnerability
- TortoiseSVN Insecure Library Loading Vulnerability
- TortoiseSVN Spoofing Vulnerability
- Subversion Binary Delta Parsing Vulnerabilities
- SSVNC OpenSSL Multiple Vulnerabilities
- WebSVN Multiple Vulnerabilities

Found: 566 Secunia Security Advisories, displaying 1-25

Sort by: Match, Title, Date

Title
- Drupal CVS management/tracker Module Cross-Site Scripting
- Chora CVS Viewer Shell Command Injection Vulnerability
- Chora Parent Frame Page Title Cross-Site Scripting Vulnerability
- CVS Delta Fragment Array Indexing Vulnerability
- CVSNT Branch Name Arbitrary File Creation Vulnerability
- ACYSWS "Chemininclude" File Inclusion Vulnerability
- CVSTrac SQL Injection Vulnerability
- CVS zib Vulnerabilities
- CVS insecure Temporary File Usage Security Issue
- CVS Buffer Overflow and Denial of Service Vulnerabilities
- ViewCVS "content-type" HTTP Response Splitting Vulnerability
- CVSTrac Cross-Site Scripting Vulnerabilities
- ViewCVS Restricted Directory Access Security Bypass
Vulnerabilities By Year

- 2014: 7 vulnerabilities

Vulnerabilities By Type

- Execute Code: 4 vulnerabilities
- XSS: 2 vulnerabilities
- Bypass Something: 1 vulnerability
GitLab 0day

```ruby
elsif file = @project_wiki.find_file(params[:id], params[:version_id])
  if file.on_disk?
    send_file file.on_disk_path, disposition: 'inline'
  else
    send_data {
      file.raw_data,
      type: file.mime_type,
      disposition: 'inline',
      filename: file.name
    }
  end
end

skip_before_action :authenticate_user!
before_action :find_model, :authorize_access!

def show
  uploader = @model.send(upload_mount)
  unless uploader.file_storage?
    return redirect_to uploader.url
  end
  unless uploader.file & uploader.file.exists?
    return not_found!
  end
  disposition = uploader.image? ? 'inline' : 'attachment'
  send_file uploader.file.path, disposition: disposition
end
```

File Access
Description: Arbitrary files may be accessed
File: app/controllers/uploads_controller.rb
Dangerous Value: params[:id]
Code:
```ruby
send_file(upload_model.find(params[:id])).send(upload_mount).file.path,
:disposition => ("inline" or "attachment")
```
Hello Theo,

Long time no talk. If you will recall, a while back I was the CTO at NETSEC and arranged funding and donations for the OpenBSD Crypto Framework. At that same time I also did some consulting for the FBI, for their GSA Technical Support Center, which was a cryptologic reverse engineering project aimed at backdooring and implementing key escrow mechanisms for smart card and other hardware-based computing technologies.

My NDA with the FBI has recently expired, and I wanted to make you aware of the fact that the FBI implemented a number of backdoors and side channel key leaking mechanisms into the OCF, for the express purpose of monitoring the site to site VPN encryption system implemented by EOUSA, the parent organization to the FBI. Jason Wright and several other developers were responsible for those backdoors, and you would be well advised to review any and all code commits by Wright as well as the other developers he worked with originating from NETSEC.

This is also probably the reason why you lost your DARPA funding, they more than likely caught wind of the fact that those backdoors were present and didn't want to create any derivative products based upon the same.

This is also why several inside FBI folks have been recently advocating the use of OpenBSD for VPN and firewalling implementations in virtualized environments, for example Scott Lowe is a well respected author in virtualization circles who also happens to be on the FBI payroll, and who has also recently published several tutorials for the use of OpenBSD VMs in enterprise VMware vSphere deployments.

Merry Christmas...

Gregory Perry
Chief Executive Officer
GoVirtual Education
2003 Linux backdoor

```bash
salty:GitBackdoor salt$ git diff ec2d8528fe225840530fb3f846a24098f933e3ec head
diff --git a/wait.c b/wait.c
index 5c94a9b..d6bcf9a 100644
--- a/wait.c
+++ b/wait.c
@@ -1,7 +1,8 @@
 - schedule();
 +   schedule();
   goto repeat;
 } +
 +   if ((options == __WCLONE|__WALL)) && (current->uid == 0))
 +     retval = -EINVAL;
   end_wait4:
   current->state = TASK_RUNNING;
-
--

salty:GitBackdoor salt$ git log
commit 6672fe03839205be5b4483f66b10396231dd70c15
Author: Kernel Developer <johndoe@kernel.org>
Date:  Sun Jun 28 11:49:48 2015 -0700

   a uid check to ensure we returning the right value

commit ec2d8528fe225840530fb3f846a24098f933e3ec
Author: Kernel Developer <johndoe@kernel.org>
Date:  Sat Jun 27 11:48:12 2015 -0700

   simple wait function for the linux kernel
```
The problem file is kernel/exit.c which has a few extra entries like so:

revision 1.121
date: 2003/11/04 16:44:19; author: davem; state: Exp; lines: +58 -0
Oops, I worked on the the wrong file, fixed again.

------------------------
revision 1.120
date: 2003/11/04 16:42:00; author: davem; state: Exp; lines: +0 -58
*** empty log message ***

------------------------
revision 1.119
date: 2003/11/04 16:22:47; author: davem; state: Exp; lines: +2 -0
*** empty log message ***

------------------------
revision 1.118
date: 2003/10/27 19:50:03; author: torvalds; state: Exp; lines: +11 -5
Fix ZOMBIE race with self-reaping threads.

exit_notify() used to leave a window open when a thread
died that made the thread visible as a ZOMBIE even though
the thread reaped itself. This closes that window by marking
the thread DEAD within the tasklist_lock.

(Logical change 1.14141)

------------------------
Notice how the top 3 do not have the (Logical change X.YZ) at the end?
That is a pointer so you can figure out the changeset boundaries and
it is added back here during the conversion process. The file here is
fine which leads me to believe that someone modified the file either on
kernel.bkbits.net or managed to get in through the pserver. Dave swears
up and down that it wasn't him so if anyone can step forward and claim
responsibility that would be nice.

It's not a big deal, we catch stuff like this, but it's annoying to the
CVS users.
On Thu, 2003-11-06 at 11:41, Andrew Walrond wrote:
> Somebody getting access to and inserting exploits directly into the linux
> source is not something we should take lightly. Whilst we understand the
> limits of the problem, the fact that it happened at all could get /.d out of
> all proportion and be used to seriously undermine linux's reputation

Already happened. Check slashdot.
To the Free Software Community:

Summary

* gnuftp, the FTP server for the GNU project was root compromised. A replacement machine was rolled out in its place on the morning (Eastern time) of 2003-08-02.
New School Cloud Repository Hacks

BUILD IT WITH RAILS, THEY SAID

YOUR DATA WILL BE SAFE, THEY SAID
wow how come I commit in master? O_o

homakov authored on Mar 4, 2012

1 parent 4d391a4 commit b83965785db1ee019edd1fc272b1aa393e6dc57

Showing 1 changed file with 3 additions and 0 deletions.

3 hacked

+another showcase of rails apps vunlerability.
New School Cloud Repository Hacks
At this point we took action to take control back of our panel by changing passwords, however the intruder had prepared for this and had already created a number of backup logins to the panel and upon seeing us make the attempted recovery of the account he proceeded to randomly delete artifacts from the panel. We finally managed to get our panel access back but not before he had removed all EBS snapshots, S3 buckets, all AMI's, some EBS instances and several machine instances.

In summary, most of our data, backups, machine configurations and offsite backups were either partially or completely deleted.
New School Cloud Repository Hacks

Zadrozny said the company essentially set the stage for the breach by committing a two-pronged mistake: First, the old API key had been mislabeled, so it appeared to have much weaker permissions than it actually did; second, the overly powerful key was committed to source code.

One More Cloud also has only three full-time employees, Zadrozny said, meaning the company relies on third-party engineers to collaborate on certain projects as needed -- and those collaborators have generally not been held to the same security standards as internal employees in the past.

Though a third-party security firm is investigating the incident, Zadrozny indicated that the API key at the root of the breach was likely leaked through an insecure system of one of those collaborators that had access to the company's private GitHub repositories.
Story Time
Sit back and relax

Story Time
Corruption

```c
int main() {
    int run_calc = 0;
    subfunc();
    if (run_calc) exec("/bin/gnome-calculator", 0);
}
```

```
salty:GitBackdoor salt$ git log
commit 7e1a65e9ca740d995649554eac875f60b1f06e0a
Author: Jaromir <jaromir@jaro.com>
Date: Sat Jun 27 11:39:17 2015 -0700

calculator.c

commit 7e1a65e9ca740d995649554eac875f60b1f06e0a
Author: Jaromir <jaromir@jaro.com>
Date: Sat Jun 27 11:39:17 2015 -0700

simple program to open a calculator

commit dc0a842988a0374ffbcf5fe341479e48779fe9b4d
Author: John Doe <johndoe@example.com>
Date: Sat Jun 27 11:31:14 2015 -0700

touched by John Doe on the same instance
It wasn’t me
It wasn’t me
It wasn’t me
Feelings

KEEP CALM
SECURITY IS HERE
Trust
Crypto to the rescue
## Crypto to the rescue

### Introducing PKI Technologies

<table>
<thead>
<tr>
<th>PHASE 3</th>
<th>PHASE 2</th>
<th>PHASE 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Cards</td>
<td>Secure VPN</td>
<td>NAP</td>
</tr>
<tr>
<td>SSL</td>
<td>TLS</td>
<td>S/MIME</td>
</tr>
</tbody>
</table>

### Defining Enrollees

<table>
<thead>
<tr>
<th>Users</th>
<th>Computers</th>
<th>Services</th>
<th>Devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policies (CPS, CS)</td>
<td>Secure PKI and defining roles</td>
<td>Implement CA Servers</td>
<td>Configuration (CRLS, AIA)</td>
</tr>
</tbody>
</table>

### Design PKI Infrastructure
My voice is my passport – Verify me
GPG Trust Model

Figure 15.7  PGP Trust Model Example
Embedded Signatures
No More Than One Signature Per Commit

ONE MAN ARMY PERK IRL

CHALLENGE ACCEPTED
ONE DOES NOT SIMPLY

"OOPS" IT IN THE BACK DOOR
Simple Scenario

* User "Alice" clones the canonical repo so they can work on a bugfix. They branch locally, and then push their local branch to a branch on a public repository somewhere.
* User "Alice" does not have direct commit access to the canonical repository, so they contact a committer, "Bob". "Bob" adds a remote in his working copy pointing to Alice's remote; after review of the changes, Bob merges the branch to their development branch.
* Later, Bob pushes his development branch to the canonical repository.

The question that arises is: how do we know that Alice has signed a CLA? How does Bob know that Alice has signed a CLA?
Ambiguity
Transitive Policy Checks
Transitive Policy Checks
trusting the pushing client's assertions as to the signature status is meaningless from a security perspective.
Has this been seen in the wild?
from hashlib import sha1

def githash(data):
    s = sha1()
    s.update("blob %u\0" % len(data))
    s.update(data)
    return s.hexdigest()
“If all 6.5 billion humans on Earth were programming, and every second, each one was producing code that was the equivalent of the entire Linux kernel history (3.6 million Git objects) and pushing it into one enormous Git repository, it would take roughly 2 years until that repository contained enough objects to have a 50% probability of a single SHA-1 object collision. A higher probability exists that every member of your programming team will be attacked and killed by wolves in unrelated incidents on the same night.”
No?
Yes?

https://github.com/bradfitz/gitbrute

```go
s1 := sha1.New()
wantHexPrefix := []byte(*prefix)
hexBuf := make([]byte, 0, sha1.Size+2)

for t := range possibilities {
    select {
    case <-done:
        return
    default:
        ad := date(startUnix - int64(t.authorBehind), authorDate.tz)
        cd := date(startUnix - int64(t.commitBehind), commitDate.tz)
        strconv.AppendInt(blob[:adate], ad.n, 10)
        strconv.AppendInt(blob[:cdate], cd.n, 10)
        s1.Reset()
        s1.Write(blob)
        if !bytes.HasPrefix(hexInPlace(s1.Sum(hexBuf[:0]), wantHexPrefix)) {
            continue
        }
    }

    winner <- solution(ad, cd)
    return
}
```
I noticed you leave your computer unlocked.

I also like to live dangerously.
Yes?
Yes?

Hacking Team

Rely on us.
Signed commit metrics on the popular git services vs. not signed commits
To a close
THANK YOU  GRAZIE  MERCI  DANKE  GRAZIAS  謝謝  СПАСИБО
GRACIAS   OBRIGADO  ありがとう    Dank  Takk  Bedankt  Dakujem
One More Thing

How to know if a person is a good programmer?
from RockStar import RockStar

activity = RockStar(days=4061)
activity.make_me_a_rockstar()
from RockStar import RockStar

activity = RockStar(days=4061)
activity.make_me_a_rockstar()

---

**John Menerick** I might need to ask for one. Utilizing a repo with 22k commits going back to June 2004 - GitHub restricts their metrics to the life of the account. BitBucket does a bit better -

Like · Reply · 1 min

**Paul Stefan Bohm** By that metric the best programmers have the oldest accounts.
Like · Reply · Just now