

Google™



Compliance and Security in the Cloud

Essential Administrative APIs for Google Apps Businesses

Michael Manoochehri
May 11th, 2011

Compliance and Security in the Cloud

Essential Administrative APIs for Google Apps Businesses

Hashtags: [#io2011](#) [#googleapps](#)

Feedback: <http://goo.gl/zCT1M>

About Me



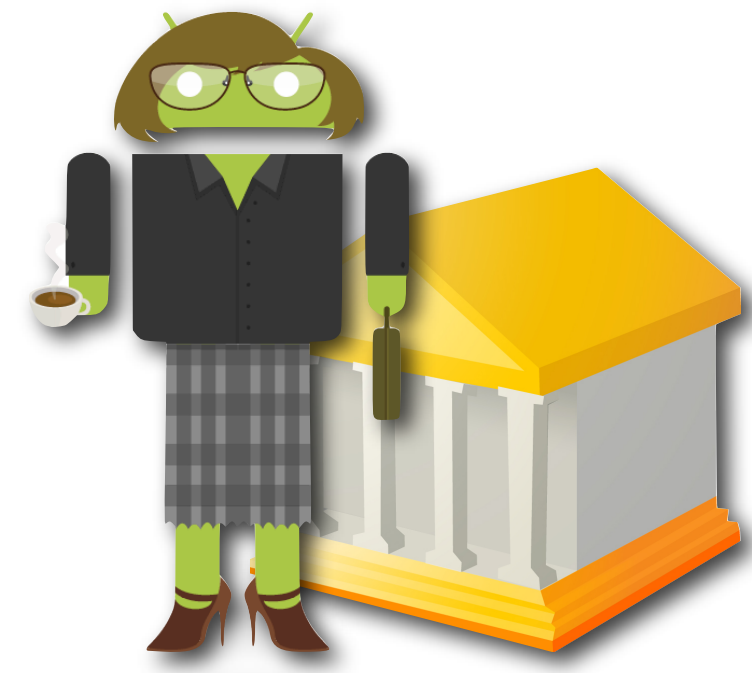
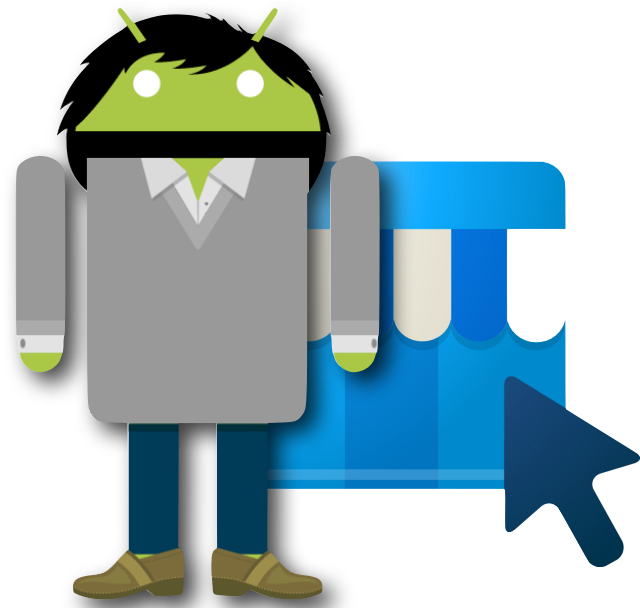
- Michael Manoochehri
- Developer Relations team supporting Google Apps APIs
- @nTangledMichael

Who Are You?

- Google Apps domain administrators
- Enterprise information managers
- System Admins from government and educational institutions
- Google Apps resellers, ISVs and Systems Integrators

Who Are You?

- Google Apps domain administrators
- Enterprise information managers
- System Admins from government and educational institutions
- Google Apps resellers, ISVs and Systems Integrators



What This Session Will Cover

- Compliance and security challenges
- How Google Apps adds value to enterprise IT
- Introduce the Google Apps administrative APIs
- APIs in practice: common use cases and code examples
- Q & A

What This Session Will Cover

- This is a 201 level session, so we assume technical background
- We will be covering a broad range of technologies
- Reminder: this session covers technologies available for Business, Education, Government, Non-Profit, and ISP editions of Google Apps

How Google Apps Adds Value



Productivity and Innovation

Realtime collaboration, constant updates, new features



Platform Independence

Work anywhere from any computer or mobile device



Reduced IT Complexity

Least complex, least expensive to license and manage

How Google Apps Adds Value



Security and Availability

Same uptime and infrastructure used for Google products



Built-in Enterprise Security Features

2-Factor Authentication, Single Sign On, Reporting Tools



Compliance Regulations Can Change Fast



Michael Oxley



Paul Sarbanes

Source: <http://en.wikipedia.org/wiki/File:SarbanesOxley.jpg>

Compliance Regulations Can Change Fast

- SOX led to a change in regulations
 - Stricter rules around auditor independence
 - New financial reporting requirements
 - Audits of organizational controls
- Regulations differ from country to country

Fiscal Compliance *Includes* Information Compliance

- What do your internal reporting and control systems look like?
- Do you need to establish compliance for a standards certification?
- How much does it cost to audit your information assets?
- What are some risks associated with user data?
 - Privacy issues
 - Confidentiality issues
 - Employees leaking information
 - Security breaches
 - etc.

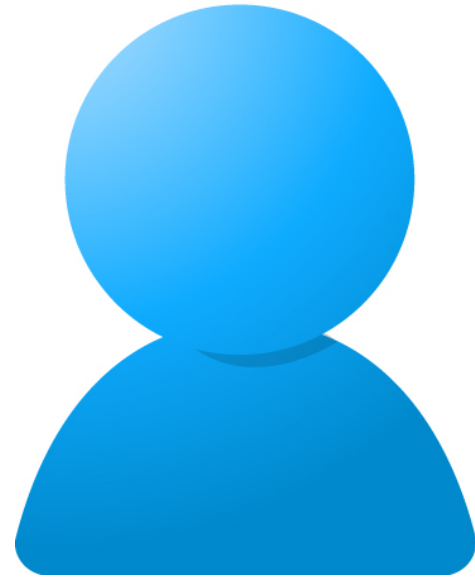
Day to Day Security Challenges

- Are my users' passwords safe?
- How can I tell if an account has been compromised?
- How do I deal with identity management?

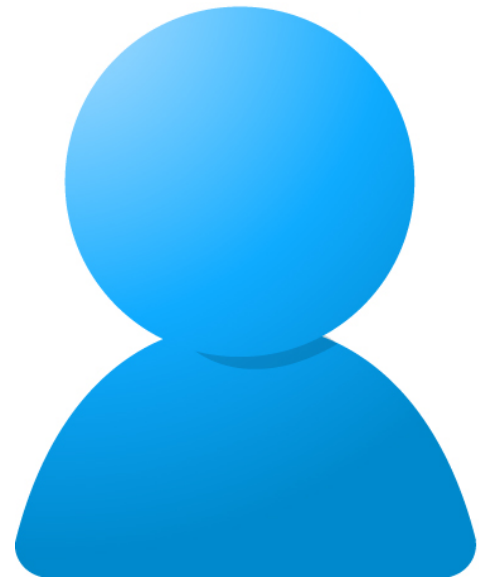
1. Dealing with compliance and security challenges often mean being *prepared for the unknown or unexpected*

1. Dealing with compliance and security challenges often mean being *prepared for the unknown or unexpected*
2. The Google Apps Administrative APIs are a set of tools for addressing these types of challenges

Google Apps Administrative APIs



Google Apps Administrative APIs



Administrators



Users

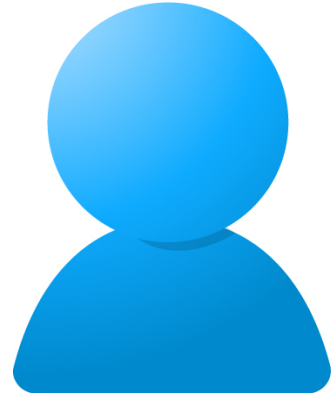


Institutions

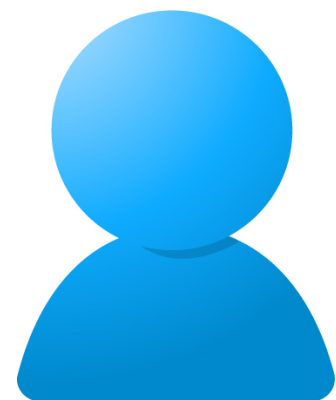


Documents/Services

Google Apps Administrative APIs



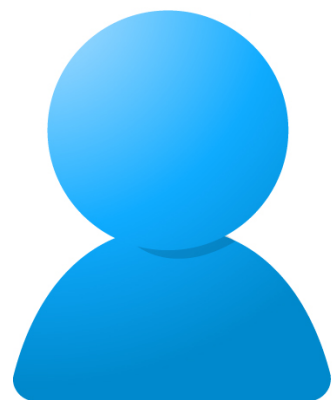
Google Apps Administrative APIs



Provisioning API
Email Migration API



Google Apps Administrative APIs

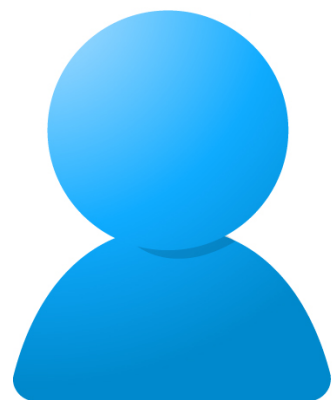


Provisioning API
Email Migration API

Admin Settings API
Email Settings API



Google Apps Administrative APIs



Provisioning API
Email Migration API

Admin Settings API
Email Settings API



Audit API
Reporting API

Why Google Apps Administrative APIs?

The screenshot displays the Google Apps Admin console interface for the domain 'upstartstartup.biz'. The user 'Michael Manoochchri' is selected, and the 'User information' tab is active. The interface shows various settings for the user, including general information, password settings, password strength, contact sharing, two-factor authentication, email quota, nicknames, groups, and email routing. The 'General' section shows the user's name as 'Michael Manoochchri' and email as 'admin@upstartstartup.biz'. The 'Password' section shows the password was set by the user and includes options to require a change or reset sign-in cookies. The 'Password Strength' section shows a strength rating of 9 characters. The 'Contact sharing' section is checked. The 'Two-factor Authentication' section is OFF. The 'Email quota' section shows 0% of 25 GB. The 'Nicknames' section shows a temporary email address. The 'Groups' section shows the user is a member of the 'test-group@upstartstartup.biz' group. The 'Email routing' section shows the user's email routing settings.

| Category | Value / Action |
|---------------------------|--|
| General | Michael Manoochchri Change name admin@upstartstartup.biz Last signed in Apr 8 |
| Password | Set by user Change password <input type="checkbox"/> Require a change of password in the next sign in Reset sign-in cookies Reset cookies and prompt the user to sign in (includes desktop and mobile devices) Learn more |
| Password Strength | When the user last logged in, their password was 9 characters long. It was also given a strength rating of |
| Contact sharing | <input checked="" type="checkbox"/> Automatically share Michael's contact information when contact sharing is enabled. |
| Two-factor Authentication | OFF Users are not allowed to turn on two-factor authentication. Learn more |
| Email quota | 0% of 25 GB |
| Nicknames | admin @upstartstartup.biz.test-google-a.com (temporary email) Add a nickname A nickname is another address where people can email Michael. |
| Groups | This user is a member of the following groups: test-group@upstartstartup.biz Edit group membership |
| Email routing | Email routing begins once you start delivering email to Google's servers. Learn more about mail routing. You can assign individual destinations for this user. Destination Change SMTP envelope |

Why Google Apps Administrative APIs?

The screenshot displays the Google Apps Admin console interface for the domain 'upstartstartup.biz'. The user 'Michael Manoochchri' is selected, and the 'User information' tab is active. The interface shows various user settings and their current states:

| Category | Value / Action |
|---------------------------|--|
| General | Michael Manoochchri Change name admin@upstartstartup.biz Last signed in Apr 8 |
| Password | Set by user Change password <input type="checkbox"/> Require a change of password in the next sign in Reset sign-in cookies Reset cookies and prompt the user to sign in (includes desktop and mobile devices) Learn more |
| Password Strength | When the user last logged in, their password was 9 characters long. It was also given a strength rating of |
| Contact sharing | <input checked="" type="checkbox"/> Automatically share Michael's contact information when contact sharing is enabled. |
| Two-factor Authentication | OFF Users are not allowed to turn on two-factor authentication. Learn more |
| Email quota | 0% of 25 GB |
| Nicknames | admin @upstartstartup.biz.test-google-a.com (temporary email) Add a nickname A nickname is another address where people can email Michael. |
| Groups | This user is a member of the following groups: test-group@upstartstartup.biz Edit group membership |
| Email routing | Email routing begins once you start delivering email to Google's servers. Learn more about mail routing. You can assign individual destinations for this user. Destination Change SMTP envelope |

At the bottom of the page, there are links for [Terms of Service](#), [Privacy policy](#), [Suggest a feature](#), and [Google Home](#), along with the copyright notice ©2011 Google Inc.

Why Google Apps Administrative APIs?

The screenshot shows the Google Admin console interface for the domain `upstartstartup.biz`. The navigation menu on the left includes `upstartstartup.biz`, `Engineering`, `Sales`, and `User Experience`. The main content area is titled `upstartstartup.biz » Engineering` and has tabs for `Users` and `Services`. Below the tabs are buttons for `Create a new user` and `Move to`. A warning message with a yellow triangle icon states: `API access is enabled. Any up...`. Below the warning is a table with a header row containing a checkbox and the text `Name`. The table lists three entries: `Michael Manoochehri`, `page larry`, and `google mr`, each with a checkbox to its left.






Why Google Apps Administrative APIs?

[upstartstartup.biz](#) » [Engineering](#) » **Michael Manoochehri**

[User information](#) **Resolved settings** [Privileges](#)

Below are services this user can sign into based on the inherited settings from the Organizations they belong to.

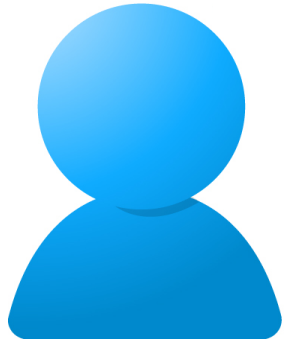
Core Google Apps suite (6)

| | |
|---|----------------|
|  Gmail | Enabled |
|  Google Calendar | Enabled |
|  Google Docs | Enabled |
|  Google Groups for Business | Enabled |
|  Google Sites | Enabled |

Why Google Apps Administrative APIs?

- Ability to create tools to automate complex tasks
- Allows Google Apps to adapt to custom situations
- Platform to build new software and services
- Grow a business providing services for Google Apps customers

Google Apps Administrative APIs



Available for use by domain admins

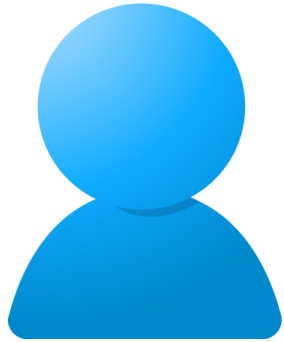


Google Apps for Business,
.edu, .gov & ISP customers



Authorization via OAuth

Google Apps Administrative APIs



Available for use by domain admins



Google Apps for Business,
.edu, .gov & ISP customers



Authorization via OAuth

More on Authorization



Identity and Data Access: OpenID and OAuth

Ryan Boyd, Google I/O 2011

Apps Admin APIs Use the Google Data Protocol

- XML-based standard derived from AtomPub
- RESTful: Create, Update, Delete Resources

Apps Admin APIs Use the Google Data Protocol

```
<?xml version='1.0' encoding='utf-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
      xmlns:gd='http://schemas.google.com/g/2005'
      gd:etag='W/"C0QBRXcycSp7ImA9WxRVFUk."'>
  <title>Foo</title>
  <updated>2011-01-23T16:25:00-08:00</updated>
  <id>http://www.example.com/someFeed</id>
  <author>
    <name>User Name</name>
  </author>
  <link href='/someFeed' rel='self' />
</feed>
```

Google Apps Administrative APIs

- Libraries and Service Classes are generally available for:
 - Java
 - .NET
 - Python
 - PHP

Google Apps Administrative APIs



Google Data Python Client Library

Administrative APIs in Practice



VARs and ISVs



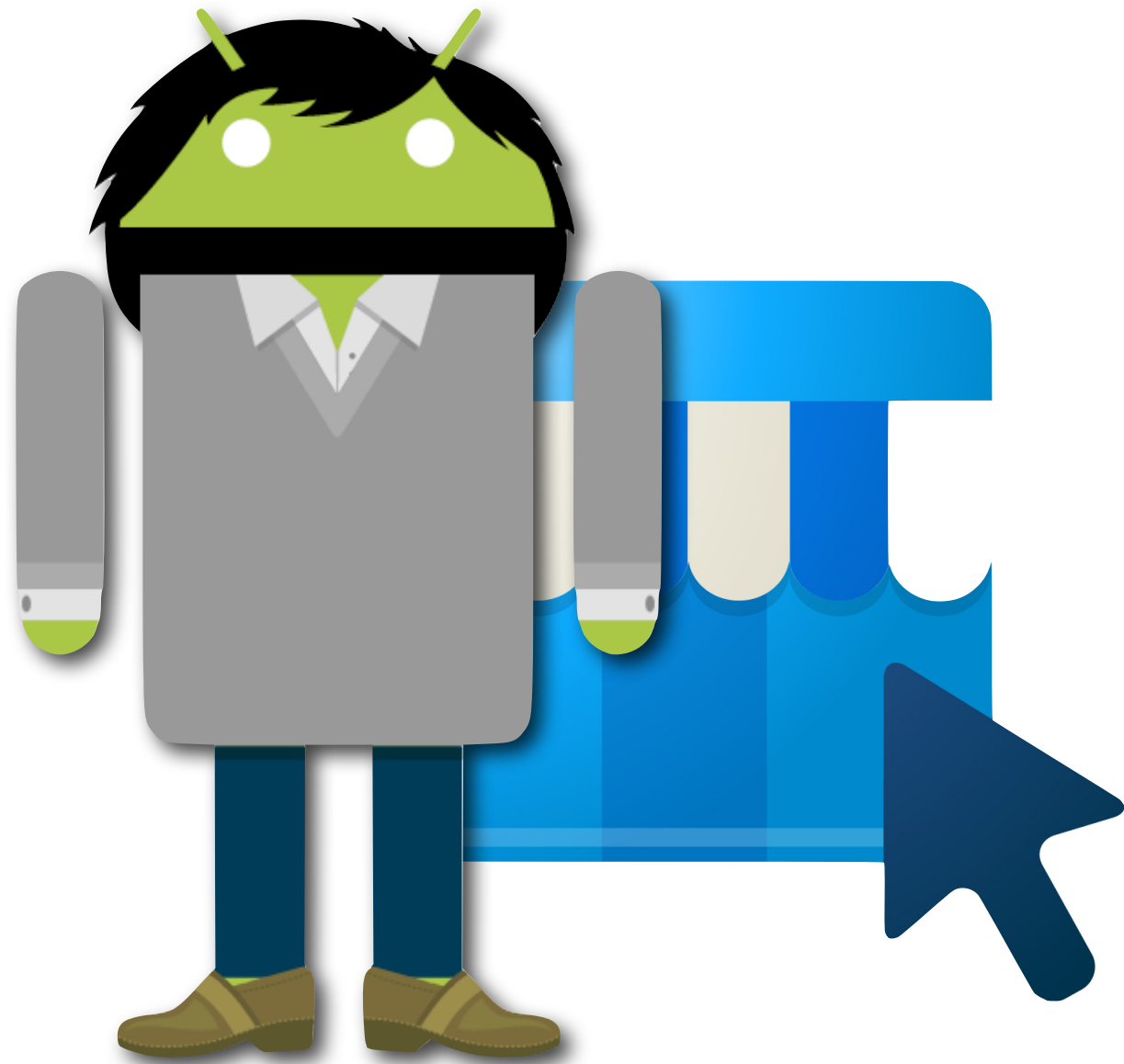
Large Institutions



Small Businesses

The Tale of the Google Apps Reseller

Meet Subradeep, a Google Apps Reseller



- Automate repetitive administrative processes
- Build tools to aid customer deployment
- Build alternative administration tools
- Reporting tools to help with billing

Migrating User Accounts with Hashed Passwords

| Username | Password | Password MD5 Hash |
|------------|----------|----------------------------------|
| susanjones | | d27117a019717502efe307d110f5eb3d |
| larryp | | 9c42a1346e333a770904b2a2b37fa7d3 |
| sergeyb | | 0da7173020b31ca78d5104fbf170ed57 |



Migrating User Accounts with Hashed Passwords

| Username | Password | Password MD5 Hash |
|------------|----------|----------------------------------|
| susanjones | ???? | d27117a019717502efe307d110f5eb3d |
| larryp | ???? | 9c42a1346e333a770904b2a2b37fa7d3 |
| sergeyb | ???? | 0da7173020b31ca78d5104fbf170ed57 |



Migrating User Accounts with Hashed Passwords

- Customer's Legacy System: MD5 hashed passwords
- How to provision accounts in Gmail quickly and securely?



Migrating User Accounts with Hashed Passwords

- Customer's Legacy System: MD5 hashed passwords
- How to provision accounts in Gmail quickly and securely?



Provisioning API

Create and Manage Google Apps Users, Groups, Organization Units



- Read/write access only available to Google Apps for Business, .edu, .gov, and ISPs customers
- Available in all product versions for read-only access by Apps Marketplace applications
- Must be enabled in the Google Apps cPanel

Provisioning API

Creating a New User Account (with an MD5 hashed password)

POST `https://apps-apis.google.com/a/feeds/yourdomain.com/user/2.0`



Provisioning API

Creating a New User Account (with an MD5 hashed password)

```
<?xml version="1.0" encoding="UTF-8"?>
<atom:entry xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:apps="http://schemas.google.com/apps/2006">
  <atom:category scheme="http://schemas.google.com/g/2005#kind"
    term="http://schemas.google.com/apps/2006#user" />
  <apps:login userName="susan_jones"
    password="d27117a019717502efe307d110f5eb3d"
    hashFunctionName="MD5" />
  <apps:name familyName="Jones"
    givenName="Susan" />
</atom:entry>
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

```
<?xml version="1.0" encoding="UTF-8"?>
<atom:entry xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:apps="http://schemas.google.com/apps/2006">
  <atom:category scheme="http://schemas.google.com/g/2005#kind"
    term="http://schemas.google.com/apps/2006#user" />
  <apps:login userName="susan_jones"
    password="d27117a019717502efe307d110f5eb3d"
    hashFunctionName="MD5" />
  <apps:name familyName="Jones"
    givenName="Susan" />
</atom:entry>
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

Step 1: Import the AppsService Class

```
import gdata.apps.service
```

```
provisioning = gdata.apps.service.AppsService()
```

```
# (Authorize)
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

Step 2: Call the CreateUser Method

```
provisioning.CreateUser(  
    'susan_jones',  
    'Jones',  
    'Susan',  
    'd27117a019717502efe307d110f5eb3d',  
    password_hash_function='MD5')
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

Step 2: Call the CreateUser Method

```
provisioning.CreateUser(  
    'susan_jones',  
    'Jones',  
    'Susan',  
    'd27117a019717502efe307d110f5eb3d',  
    password_hash_function='MD5')
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

List of Account Information...

```
accounts = [  
    ('susanjones', 'Susan', 'Jones', 'd27117a019717...'),  
    ('larryjoe', 'Larry', 'Joe', '5442b8bccadc6...'),  
    #etc...  
]
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

Iterate over the List

```
for (username, first, last, MD5password) in accounts:  
  
    provisioning.CreateUser(  
        username,  
        first,  
        last,  
        MD5password,  
        password_hash_function='MD5' )
```



Provisioning API

Creating a New User Account (with an MD5 hashed password)

Iterate over the List

```
for (username, first, last, MD5password) in accounts:
```

```
    provisioning.CreateUser(  
        username,  
        first,  
        last,  
        MD5password,  
        password_hash_function='MD5' )
```

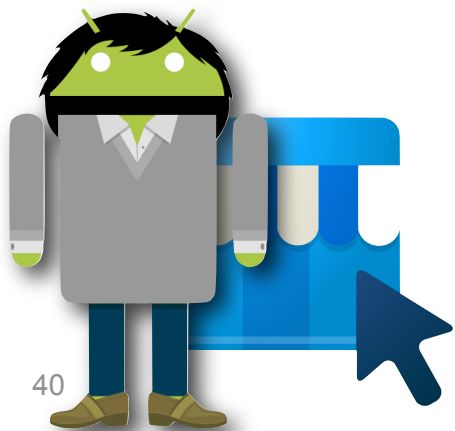


Provisioning API

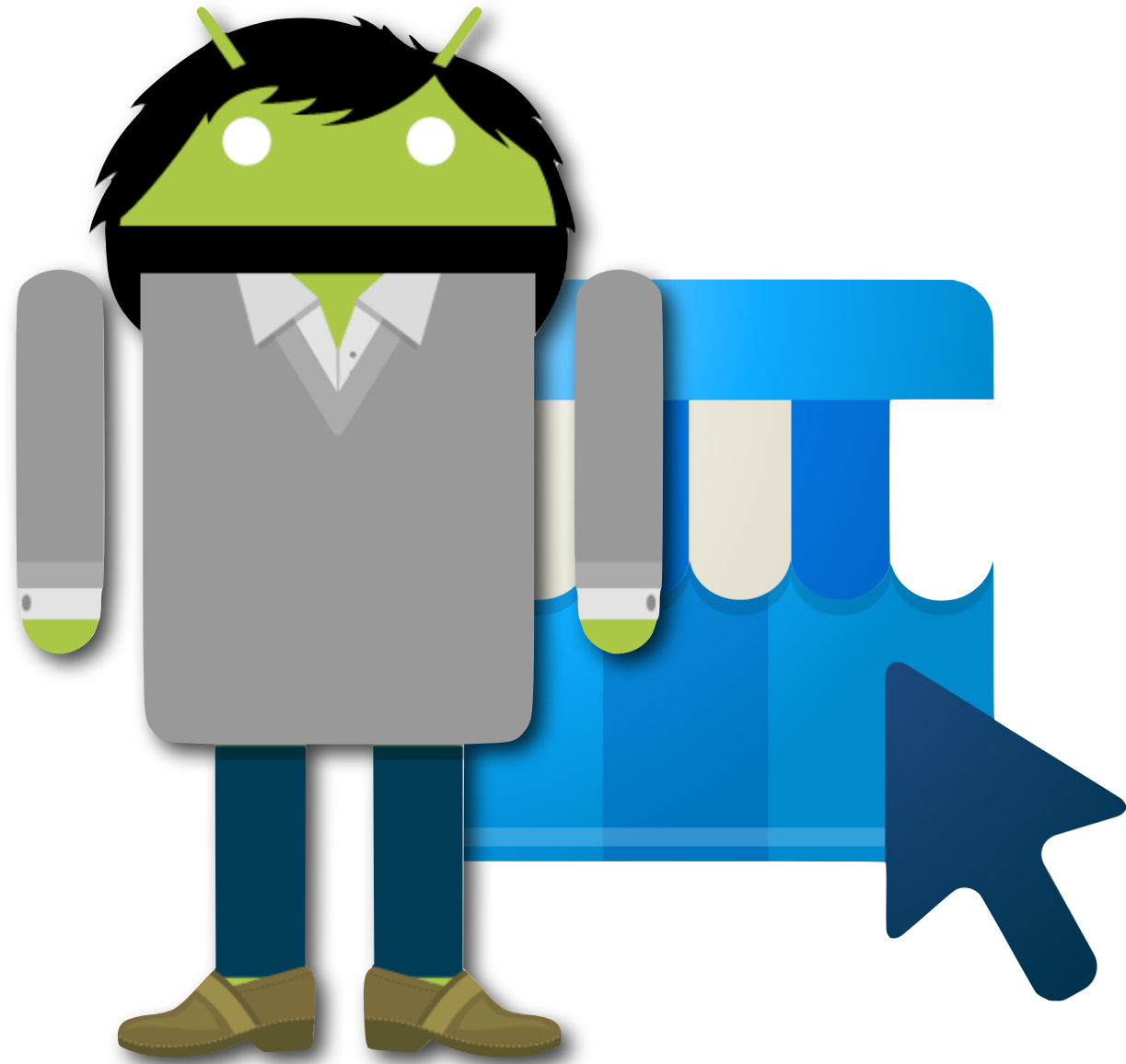
Creating a New User Account (with an MD5 hashed password)

Iterate over the List

```
for (username, first, last, MD5password) in accounts:  
  
    provisioning.CreateUser(  
        username,  
        first,  
        last,  
        MD5password,  
        password_hash_function='MD5')
```



Subradeep Needs to Create Many Users Quickly



- Rate limits aren't definite
- We recommend an exponential or linear back-off approach

Provisioning API

Creating Many User Accounts Quickly

Step 1: Import time and Exception Classes

```
import time  
from gdata.apps.service import AppsForYourDomainException
```

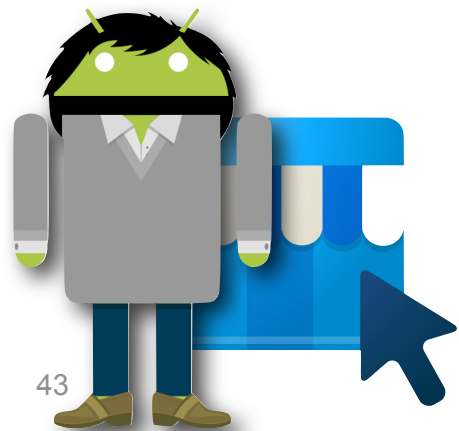


Provisioning API

Creating Many User Accounts Quickly

Step 2: Variables to Keep Track of Delay

```
delay = 0  
delay_increment = 5
```



Provisioning API

Creating Many User Accounts Quickly

Step 3: Handle Provisioning API Exceptions

```
for (username, first, last, MD5password) in accounts:
```

```
    provisioning.CreateUser(username, first, last,  
                            MD5password, password_hash_function='MD5')
```



Provisioning API

Creating Many User Accounts Quickly

Step 3: Handle Provisioning API Exceptions

```
for (username, first, last, MD5password) in accounts:  
    while True:  
        try:  
            provisioning.CreateUser(username, first, last,  
                                    MD5password, password_hash_function='MD5')  
            break
```

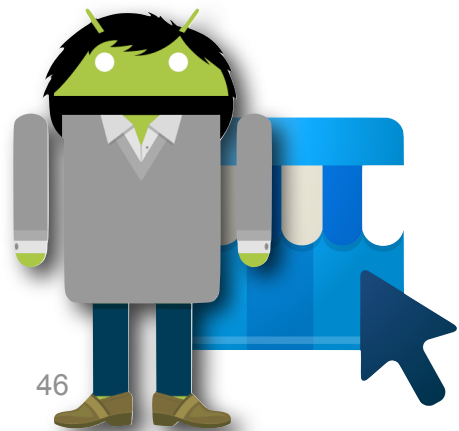


Provisioning API

Creating Many User Accounts Quickly

Step 3: Handle Provisioning API Exceptions

```
for (username, first, last, MD5password) in accounts:
    while True:
        try:
            provisioning.CreateUser(username, first, last,
                                    MD5password, password_hash_function='MD5')
            break
        except AppsForYourDomainException, e:
            if e.args[0]['status'] == 503:
                delay += delay_increment
                time.sleep(delay)
```



Provisioning API

Creating Many User Accounts Quickly

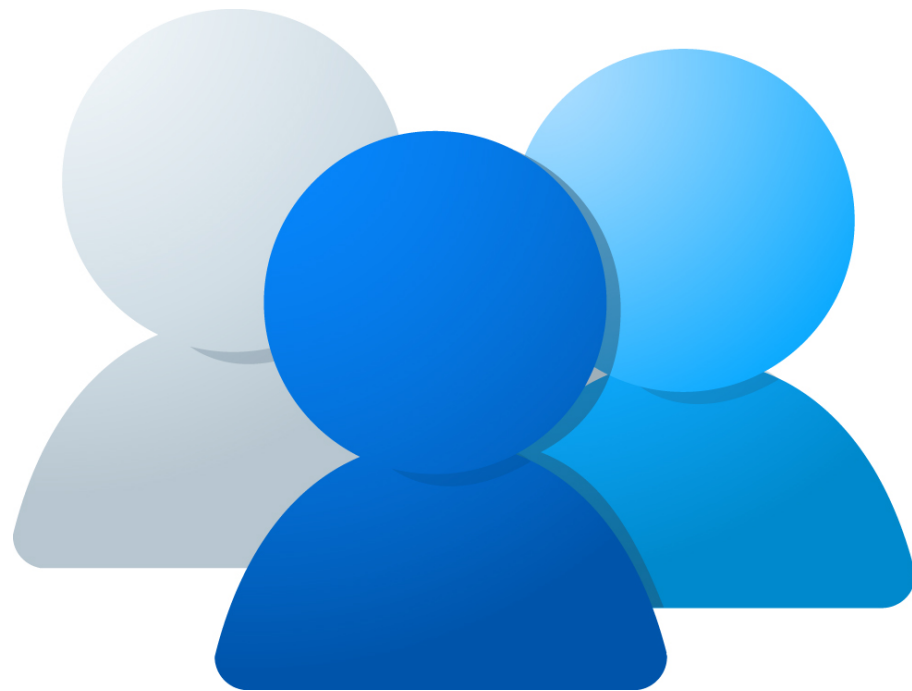
Step 3: Handle Provisioning API Exceptions

```
for (username, first, last, MD5password) in accounts:
    while True:
        try:
            provisioning.CreateUser(username, first, last,
                                    MD5password, password_hash_function='MD5')
            break

        except AppsForYourDomainException, e:
            if e.args[0]['status'] == 503:
                delay += delay_increment
                time.sleep(delay)
```

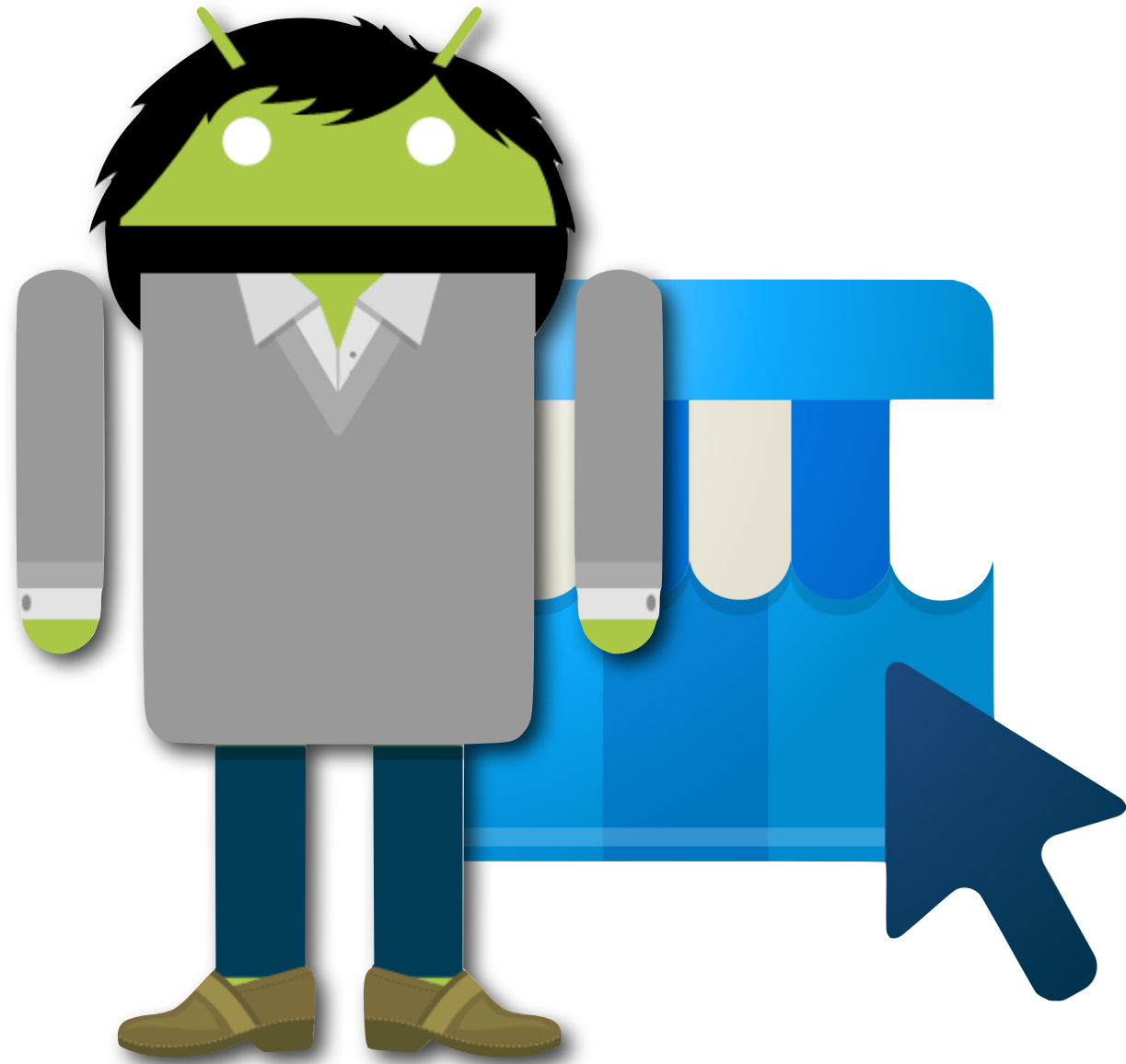


More Provisioning API Tips



- Newly Provisioned users may take time (~10 min) to appear in Control Panel
- Deleted user data is held for 5 days for security purposes
- We're working on improving throughput

Custom Migration of Legacy Email to Gmail



- What about the email?
- Google provides tools for email migration...
- ... but, sometimes those tools can't be used

Custom Migration of Legacy Email to Gmail

- Could not access customer's legacy email via IMAP due to security restrictions



50

Custom Migration of Legacy Email to Gmail

- Could not access customer's legacy email via IMAP due to security restrictions



Email Migration API

Move Your Legacy Email to the Cloud



- Add labels and properties to migrated email
- Non-Admins may use the API if an Admin has given permission
- Messages should be UTF-8 encoded

Email Migration API

Move Your Legacy Email to the Cloud



- Add labels and properties to migrated email
- Non-Admins may use the API if an Admin has given permission
- Messages should be UTF-8 encoded

Email Migration API: "RFC 2822" Messages

```
From: michael@usptartstartup.biz  
Subject: RFC 2822 message  
Date: 10 Mar 2011 09:00:28 -0700
```

```
Hi, this is an RFC2822 message body...  
for testing with the Email Migration API
```

Email Migration API

Creating a New User Account (with an MD5 hashed password)

Step 1: Import the MigrationService Class

```
from gdata.apps.migration import service

migration_service = service.MigrationService()

# (Authorize)
```



Email Migration API

Creating a New User Account (with an MD5 hashed password)

Step 2: Grab an Email (RFC 2822)

```
f = open('rfc2822mail.txt', 'r')
```

```
mail_message = f.read()
```

```
f.close()
```



Email Migration API

Creating a New User Account (with an MD5 hashed password)

Step 3: Add Labels and Properties

```
mail_item_properties = [ 'IS_INBOX', 'IS_STARRED' ]
```

```
labels = [ 'migrated' ]
```



Email Migration API

Creating a New User Account (with an MD5 hashed password)

Step 3: Add Labels and Properties

```
mail_item_properties = [ 'IS_INBOX', 'IS_STARRED' ]
```

```
labels = [ 'migrated' ]
```



Email Migration API

Creating a New User Account (with an MD5 hashed password)

Step 4: Call the ImportMail method

```
migration_service.ImportMail('michael',  
                                mail_message,  
                                mail_item_properties,  
                                labels)
```

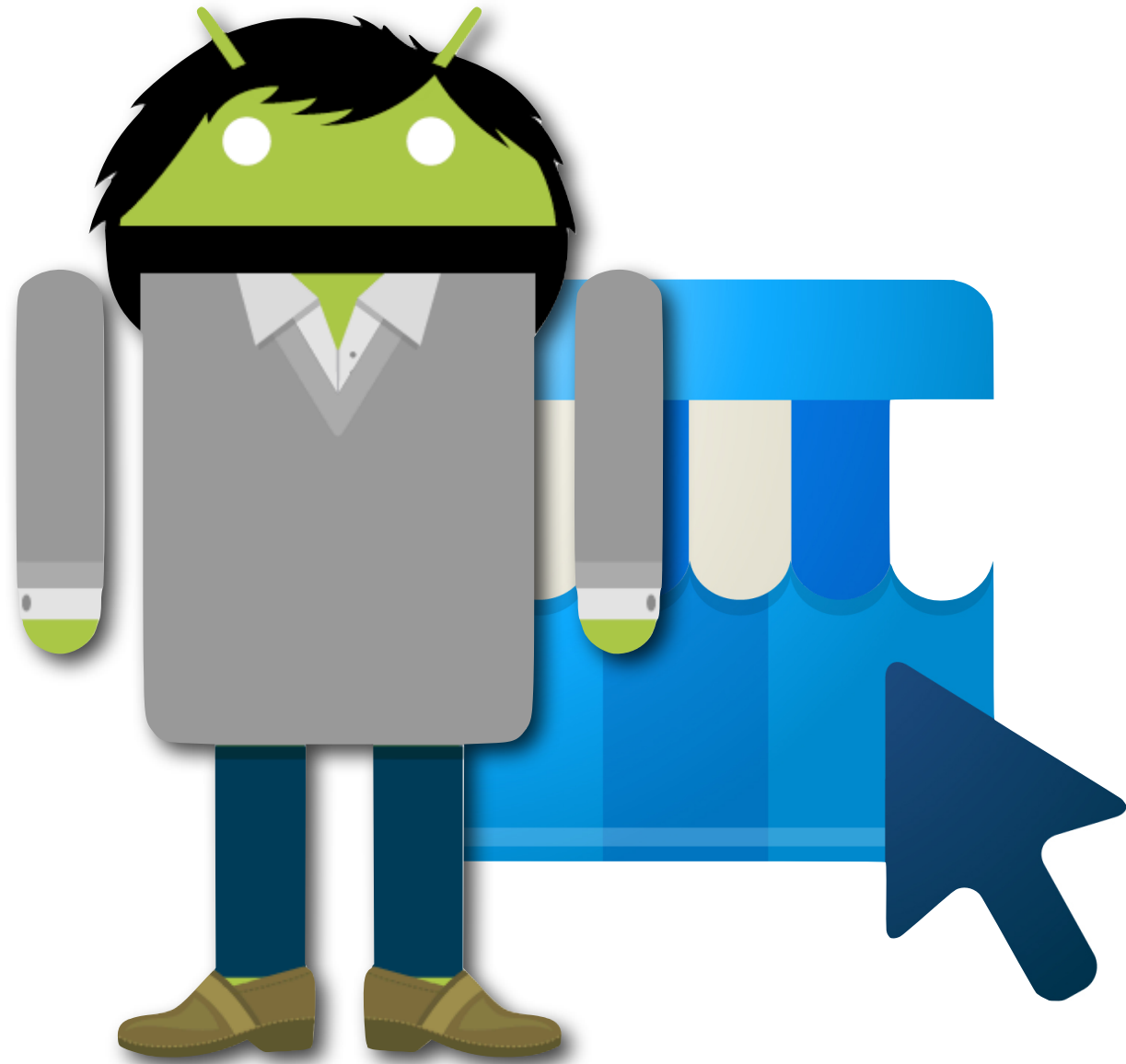


Email Migration API Tips



- The maximum size for a migrated email is 25Mb total
 - includes attachments
- No executable files

Maintaining Customer's Security Policy



Provisioning API

Create and manage users, groups, and more

Email Migration API

Migrate legacy email to the cloud

Google Apps Administrative API Toolbox



Google Apps Administrative API Toolbox



Provisioning API

Create and Update Users, Groups, Nicknames, OrgUnits



Email Migration API

Move your Legacy Email to the Cloud



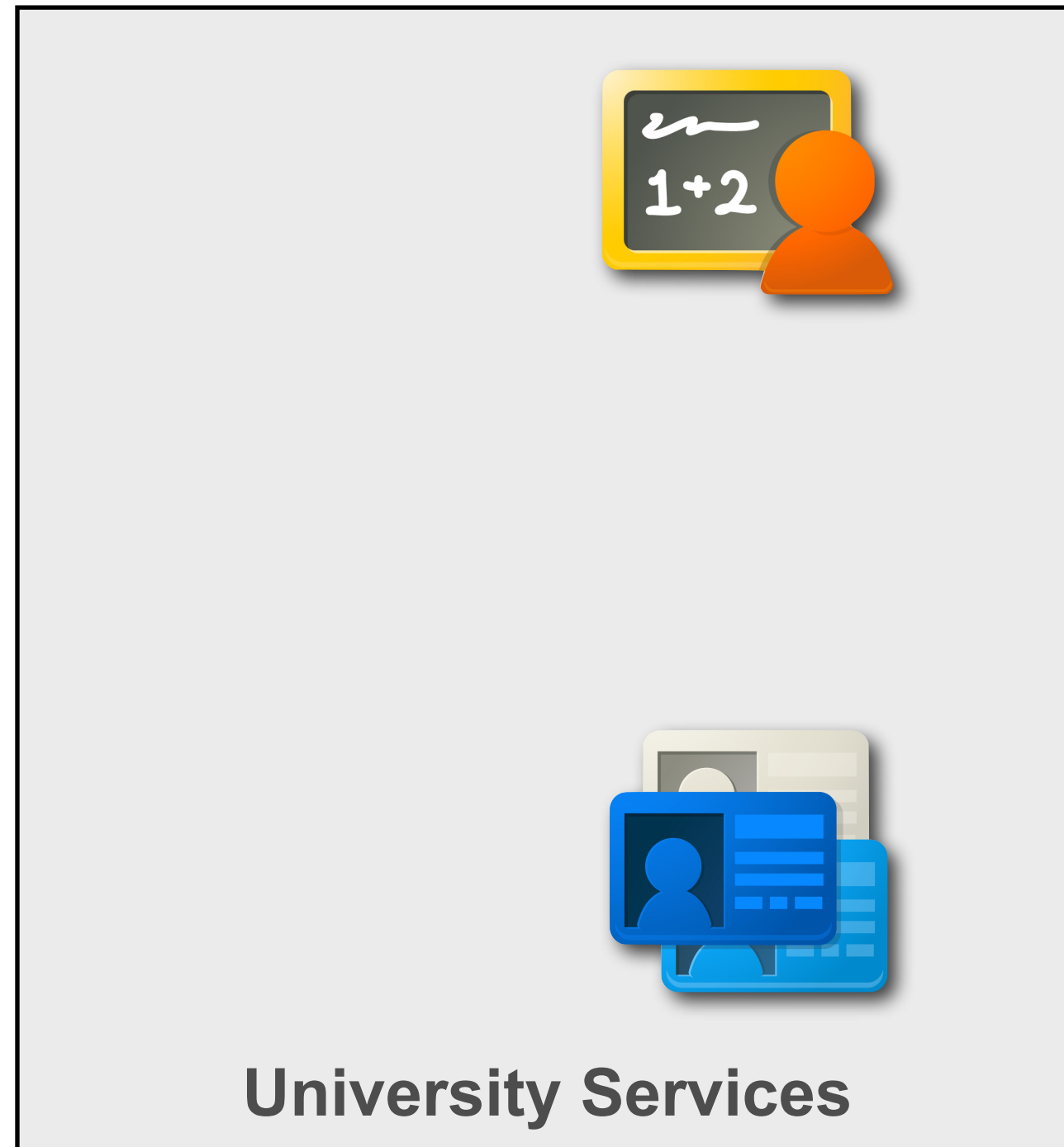
The Tale of the University Admin

Meet Oscar, System Admin at a State University



- Tens of thousands of users
- Wants to administrate user authentication via a single LDAP server
- Wants Google Apps should appear as just one of many web-based services

University Identity Management




University Identity Management



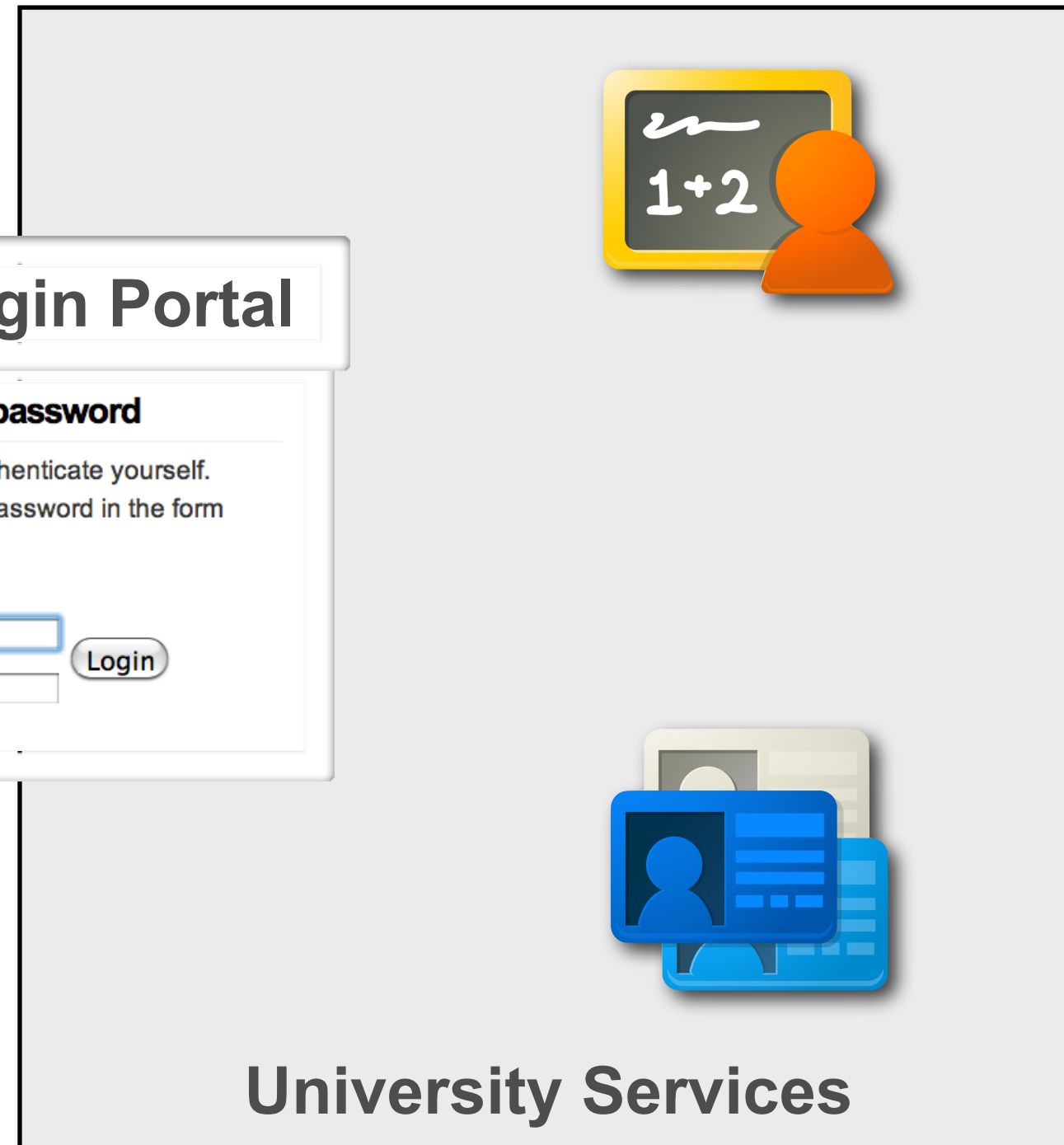
University Login Portal

Enter your username and password

A service has requested you to authenticate yourself.
Please enter your username and password in the form below.

 Username

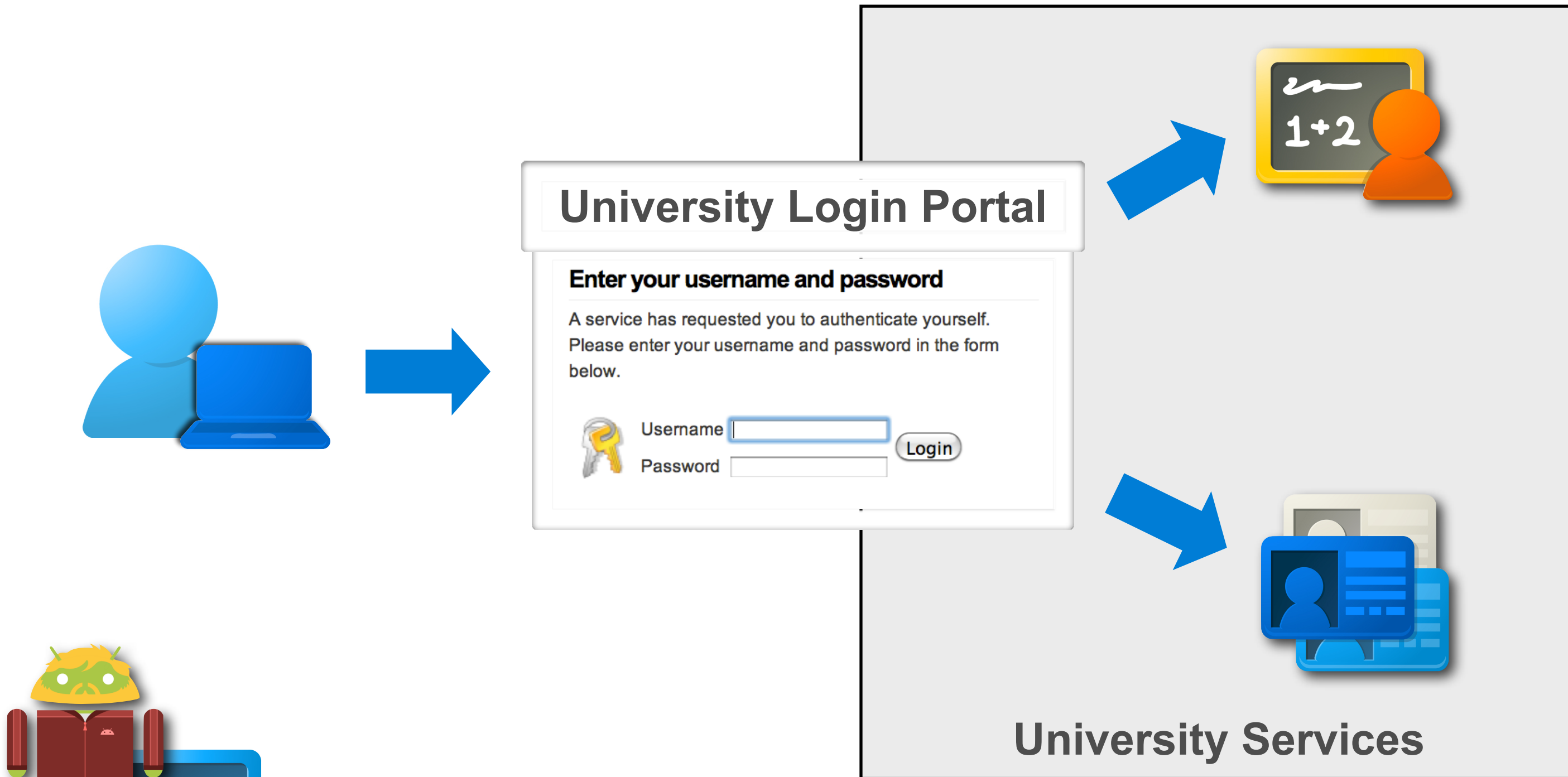
Password



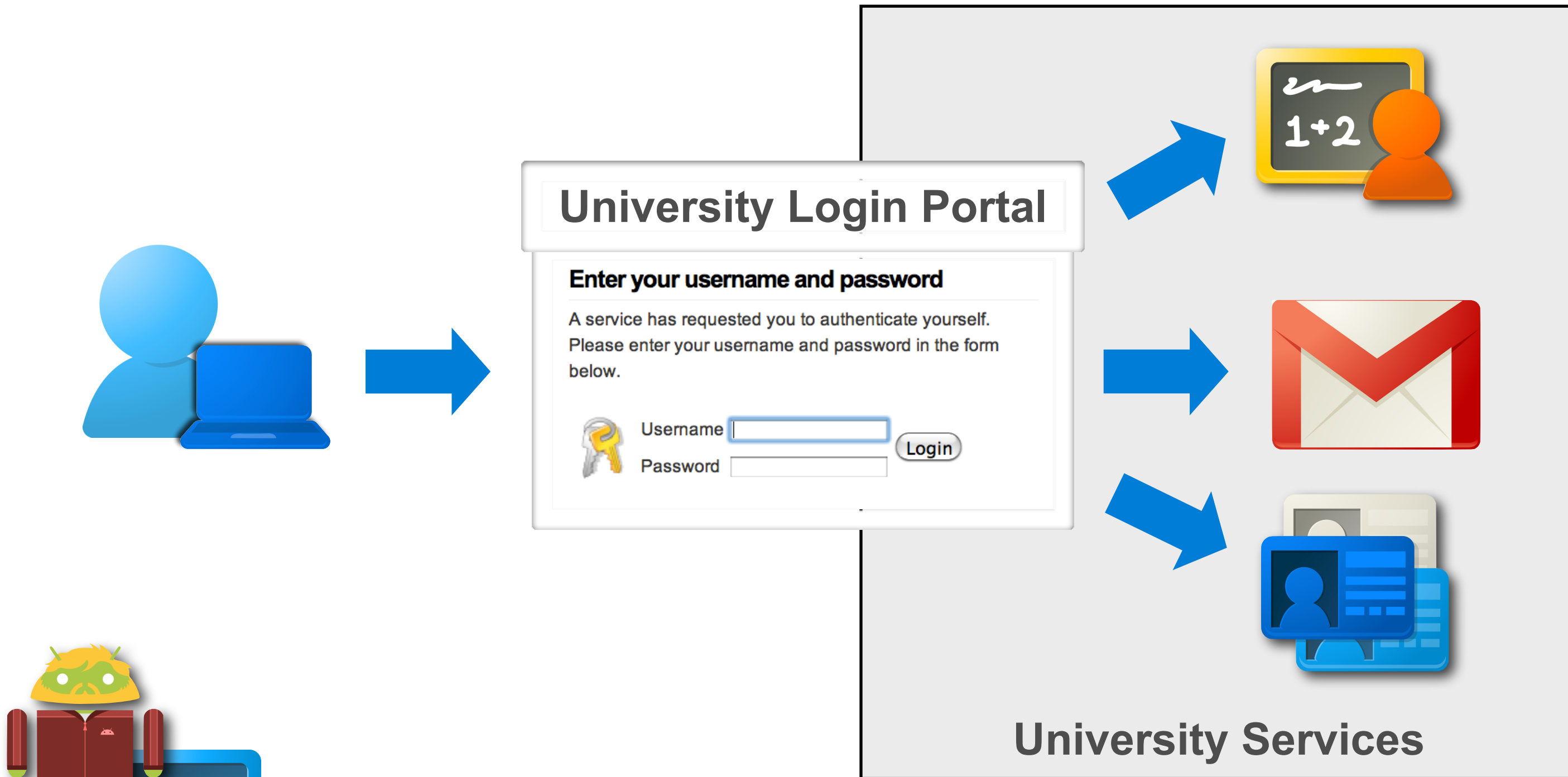
University Services

The 'University Services' interface is a light gray rectangular area. At the top right, there is a yellow-bordered icon of a chalkboard with '1+2' written on it, next to an orange person icon. At the bottom right, there is a blue icon of a person's profile next to a document icon. The text 'University Services' is centered at the bottom of this area.

University Identity Management



University Identity Management



Google Apps SAML-based Single Sign On

- Delegates all password and authentication authority to customer's Identity Provider software



Sign in with your
Google Account

Username:
ex: pat@example.com

Password:

Stay signed in

[Can't access your account?](#)



Google Apps SAML-based Single Sign On


- Delegates all password and authentication authority to customer's Identity Provider software



University Login Portal

Enter your username and password

A service has requested you to authenticate yourself. Please enter your username and password in the form below.

 Username

Password



Google Apps SAML-based Single Sign On

Dashboard Organization & users Groups Domain settings **Advanced tools** Setup Support Service settings ▾

[« Back to Advanced tools](#)

Set up single sign-on (SSO)

To set up SSO, please provide the information below. [SSO Reference](#)

Enable Single Sign-on

Sign-in page URL *
 URL for signing in to your system and Google Apps

Sign-out page URL *
 URL to redirect users to when they sign out

Change password URL *
 URL to let users change their password in your system

Verification certificate *
A certificate file has been uploaded. [Replace certificate](#)

The certificate file must contain the public key for Google to verify sign-in requests. [More information](#)

University Login Portal

Enter your username and password

A service has requested you to authenticate yourself. Please enter your username and password in the form below.

 Username

Password



Google Apps SAML-based Single Sign On

- Delegates all password and authentication authority to customer's Identity Provider software
- Adheres to the SAML 2.0 standard
- Only supports a "resource provider" initiated login flow



Finding Unused Apps Accounts



- Budget is tight - the Chancellor is conducting a financial audit
- How many University Email Accounts are actually being used?
- Are students using POP/Email Clients?
- Do we need to support alumni?

Reporting API

Generate Reports About User Activity on a Google Apps Domain



- Returns information such as:
 - Account Creation Date
 - Account Activity
 - POP/IMAP Access
- Authorization is done via ClientLogin

Reporting API

Post an Account Report Request to the API

```
<?xml version="1.0" encoding="UTF-8"?>
<rest xmlns="google:accounts:rest:protocol"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <type>Report</type>
  <token>DQAAAHQAAADgjt625094TCzxcWC4Ss-MH...</token>
  <domain>example.com</domain>
  <date>2010-08-01</date>
  <page>1</page>
  <reportType>daily</reportType>
  <reportName>accounts</reportName>
</rest>
```



Reporting API

Using the Reporting API Python Client

```
>>python reporting.py \  
    --email=admin@yourdomain.com \  
    --password=adminP@s5w0rd \  
    --report=accounts
```

(Java, .NET, PHP clients also available)



Reporting API: An Accounts Report

```
date, account_id, account_name, status, quota_in_mb,  
usage_in_bytes, primary_account_id, primary_account_name,  
creation_date, last_login_date, last_web_mail_date, surname,  
given_name, service_tier, channel, suspension_reason,  
last_pop_date, creation_time, last_login_time,  
last_web_mail_time, last_pop_time
```

```
20100808, 000000cccccc33333, "michael@c_state_u.edu", "ACTIVE",  
2048, 29057823, , , 20100801, 20100808, 20060807,  
"Manoochehri", "Michael", , , , 20100807, 2010-08-01,13:15:29,  
2010-08-08, 2010-08-08 15:12:43, 2010-08-07 19:43:12,  
2010-08-07 12:29:17
```



Reporting API: An Accounts Report

```
date, account_id, account_name, status, quota_in_mb,  
usage_in_bytes, primary_account_id, primary_account_name,  
creation_date, last_login_date, last_web_mail_date, surname,  
given_name, service_tier, channel, suspension_reason,  
last_pop_date, creation_time, last_login_time,  
last_web_mail_time, last_pop_time
```

```
20100808, 000000cccccc33333, "michael@c_state_u.edu", "ACTIVE",  
2048, 29057823, , , 20060801, 20100808, 20100807,  
"Manoochehri", "Michael", , , , 20100807, 2010-08-01,13:15:29,  
2010-08-08, 2010-08-08 15:12:43, 2010-08-07 19:43:12,  
2010-08-07 12:29:17
```



Reporting API

Generate Reports About User Activity on a Google Apps Domain



- Reporting API data is not real time
- may be up to 24 hours old
- Compress reports (using gzip)

Label Certain Messages as “Important”



- University uses an emergency alert system
- The Chancellor wants emails sent from his office to be labeled as "important"

Admin Settings and Email Settings API

Automate Google Apps Control Panel and Gmail Settings

- Allows for bulk updates of user email settings
- Useful for building reseller and deployment tools

Admin Settings and Email Settings API

Automate Google Apps Control Panel and Gmail Settings



- Allows for bulk updates of user email settings
- Useful for building reseller and deployment tools

Admin Settings API

- Programmatically set domain properties
- Set a domain's Single Sign On settings
- Set the domain's logo
- and more...



Email Settings API

- Filters (i.e. never send an important email to “spam”)
- Create Gmail labels
- Set Gmail delegation settings
- Set user signatures
- Set vacation settings
- and more...



Email Settings API

Create a Filter with the Email Settings API

Step 1: Import the Email Settings Client

```
import gdata.apps.emailsettings.client as client

email_settings = client.EmailSettingsClient(domain='yourdomain.com')

# (Authorize)
```



Email Settings API

Create a Filter with the Email Settings API

Step 2: Create an Email Filter for One User

```
email_settings.CreateFilter(username='michael',  
                             from_address='chancellor@yourdomain.com',  
                             has_the_word='Security Alert',  
                             label='important')
```



Email Settings API

Create a Filter with the Email Settings API

Step 2: Create an Email Filter for One User

```
email_settings.CreateFilter(username='michael',  
                             from_address='chancellor@yourdomain.com',  
                             has_the_word='Security Alert',  
                             label='important')
```



Email Settings API

Create a Filter with the Email Settings API

Step 3: Create a Provisioning Service

```
import gdata.apps.emailsettings.client as client
import gdata.apps.service

email_settings = client.EmailSettingsClient(domain='yourdomain.com')
provisioning = gdata.apps.service.AppsService()

# (Authorize)
```



Email Settings API

Create a Filter with the Email Settings API

Step 4: Set an Email Filter for all Users on Domain

```
userFeed = provisioning.RetrieveAllUsers()  
  
for user in userFeed.entry:  
    email_settings.CreateFilter(username=user.login.user_name,  
                               from_address='chancellor@yourdomain.com',  
                               has_the_word='Security Alert',  
                               label='important')
```



Email Settings API

Create a Filter with the Email Settings API

Step 4: Set an Email Filter for all Users on Domain

```
userFeed = provisioning.RetrieveAllUsers()
```

```
for user in userFeed.entry:
```

```
    email_settings.CreateFilter(username=user.login.user_name,  
                                from_address='chancellor@yourdomain.com',  
                                has_the_word='Security Alert',  
                                label='important')
```



Email Settings API

Create a Filter with the Email Settings API

Step 4: Set an Email Filter for all Users on Domain

```
userFeed = provisioning.RetrieveAllUsers()  
  
for user in userFeed.entry:  
    email_settings.CreateFilter(username=user.login.user_name,  
                                from_address='chancellor@yourdomain.com',  
                                has_the_word='Security Alert',  
                                label='important')
```



Email Settings API

```
import gdata.apps.emailsettings.client as client
import gdata.apps.service
from gdata.apps.service import AppsForYourDomainException

email_settings = client.EmailSettingsClient(domain='yourdomain.com')
provisioning = gdata.apps.service.AppsService()

# (authorize)

userFeed = provisioning.RetrieveAllUsers()

for user in userFeed.entry:
    try:
        email_settings.CreateFilter(username=user.login.user_name,
                                    from_address='chancellor@yourdomain.com',
                                    has_the_word='Security Alert',
                                    label='important')
    except AppsForYourDomainException, e:
        # handle error
```



Email Settings API

```
import gdata.apps.emailsettings.client as client
import gdata.apps.service
from gdata.apps.service import AppsForYourDomainException

email_settings = client.EmailSettingsClient(domain='yourdomain.com')
provisioning = gdata.apps.service.AppsService()

# (authorize)

userFeed = provisioning.RetrieveAllUsers()

for user in userFeed.entry:
    try:
        email_settings.CreateFilter(username=user.login.user_name,
                                    from_address='chancellor@yourdomain.com',
                                    has_the_word='Security Alert',
                                    label='important')
    except AppsForYourDomainException, e:
        # handle error
```



Enforcing Identity Policy for Many Users



SAML-based Single Sign On

Delegates Google Apps Identity Management

Reporting API

Produce Detailed Reports about Users

Admin & Email Settings APIs

Automate Administrative Tasks

Google Apps Administrative API Toolbox



Provisioning API

Create and Update Users, Groups, Nicknames, OrgUnits



Email Migration API

Move your Legacy Email to the Cloud



Google Apps Administrative API Toolbox



Provisioning API

Create and Update Users, Groups, Nicknames, OrgUnits



Email Migration API

Move your Legacy Email to the Cloud



Reporting API

Generate Reports about User Statistics



Admin & Email Settings APIs

The Remote Control for Administrative Settings

The Tale of the Financial Services CIO

Meet Carol, the CIO of a Medium-Sized Business

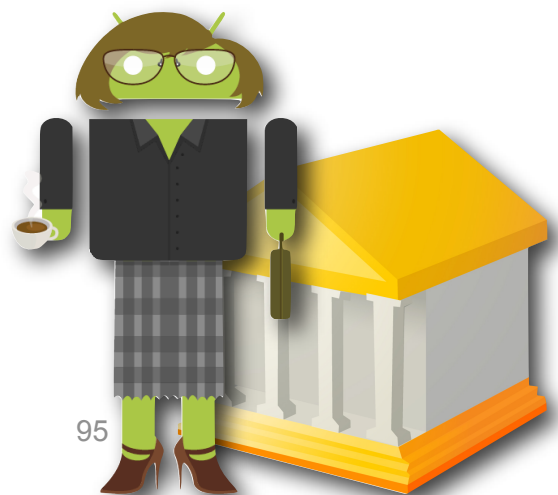


- Financial services company
- Still small, but growing fast
- 300 employees
- CIO likes the fact that doesn't need to worry about hardware and software installation

Carol Has... a Problem



- Carol thinks that one of her user's accounts has been compromised
- How to find out if the account has been accessed from a suspicious IP address?



Audit API

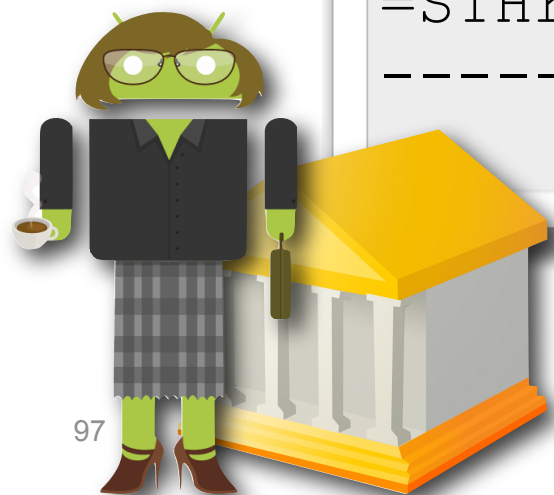


- Admins may download an encrypted file of a domain's email, chats, as well as login information
- Email Monitor: Silently forward email to another account
- Legal Compliance Purposes Only

Audit API

Uploading a OpenPGP Public Key

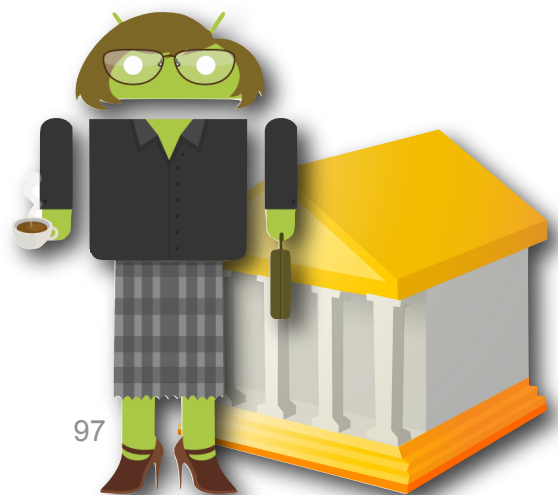
```
-----BEGIN PGP PUBLIC KEY BLOCK-----  
Version: GnuPG v1.4.10 (GNU/Linux)  
  
mQENBErWaD4BCACT2ngfs6/+QOGYbxNbc3gLnXHtqp7NTTXNW4SJo+/A1oUZoGxA  
Qx6zFXhQ/8MXW66+8STS1YqNJOARFtjbIKPwjrdcukdPzYVKGZre0RaxCnMyCV+6  
F4YNQD1UegHTu2wCGR1uiYOfLxUa7/do6s31WRTH8vbtipy9/6obEIXDjDzKIqYO  
rvRDWqALBYk10kJ3Hbgfy142EsnLiAhS+dMs2PCDi2X0ZJCPZ8eTjLsdAtqVZJ+R  
WC1J3UDuFfmcpsDYRtUL9w6YMt1apC+9mmJ3ABEBAAG0V0Rhc2hlciBUZXN0IChU  
dGVyMkYXNoZXItaHlkLXRlc3QuY29tPokBOAQTAAQIAIqUCStZoPgIbDQYLCQgH  
k19QckTpwBdsKEYumFvmWve5UX2SVV7fzOC0nZtgFxtZGlJhGmjsA3rxFTlb+Ira  
WZayXCWYiCzd7m9z5/KyGD2GFTK/94mdm25N6GXh/b35pIFZXBI/rZjrYrhYRBFu  
GtzGFIw9AAnFyUzEUUVfPWUtBe5yHmw54C60nHk5xYIa6qFhiLp4PYqZCrYX1iIs  
fRROFA==  
=STHr  
-----END PGP PUBLIC KEY BLOCK-----
```



Audit API

Uploading a OpenPGP Public Key

```
LS0tLS1CRUdJTiBQR1AgUFVCTE1DIETfWSBCTE9DSy0  
tLS0tDQpWZXJzaW9uOiBHbnVQRyB2MS40LjEwIChHT1  
UvTG1udXgpDQoNCm1RRU5CRXJXYUQ0QkNBQ3Qybmdmc  
zYvK1FPR1lieE5iYzNnTG5YSHRxcDdOVFRYT1c0U0pv  
Ky9BMW9VWm9HeEENC1F4NnpGWGhRLzhNWFc2Nis4U1R  
TMV1xTkpPQVJGdGpiSUtQd2pyZGN1a2RQe1lWS0dacm  
UwUmF4Q25NeUNWKzYnckY0WU5RRDFVZWdIVHUyd0NHU  
jF1aV1PZkx4VWE3L2RvNnMzMVdSVEg4dmJ0aVBZOS8T  
I1TjZHWGgvYjM1cE1GW1hCSS9yWmpyWXJoWVJCRnUNC  
kd0ekdGSXc5QUFuRn1VekVVVVZmUFdVdEJ1NX1ITVc1  
NEM2MG5IazV4WU1hNnFGaGlMcDRQWXFaQ3JZWDFpSXM  
NCmZSuk9GQT09DQo9U1RIcg0KLS0tLS1FTkQgUEdQIF  
BVQkxJQyBLRVkgQkxPQ0stLS0tLQ==
```

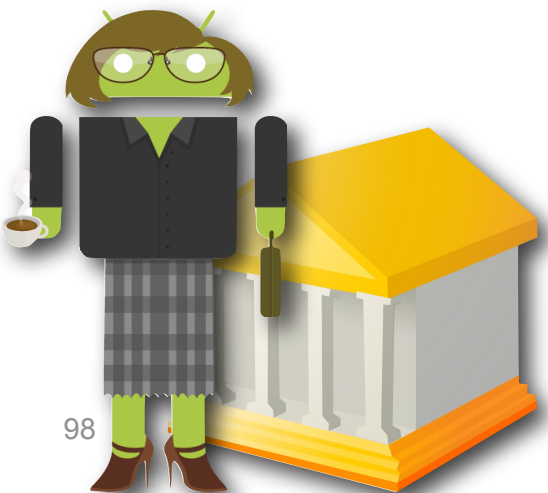


Audit API

Uploading an OpenPGP Public Key

Step 1: Import the AuditService methods

```
from gdata.apps.audit import service  
  
audit_service = service.AuditService()  
  
# (Authorize)
```



Audit API

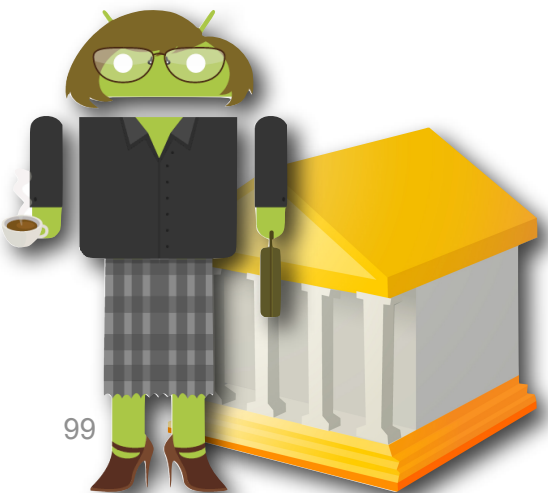
Uploading a OpenPGP Public Key

Step 2: Load the OpenPGP Key into a variable

```
f = open('PGPKey.txt', 'r')
```

```
pgpkey = f.read()
```

```
f.close()
```

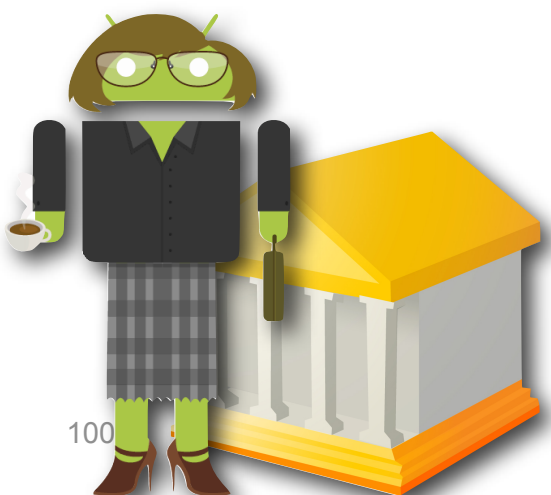


Audit API

Uploading a PGP Public Key

Step 3: Call the updatePGPKey method

```
audit_service.updatePGPKey (pgpkey)
```



Audit API



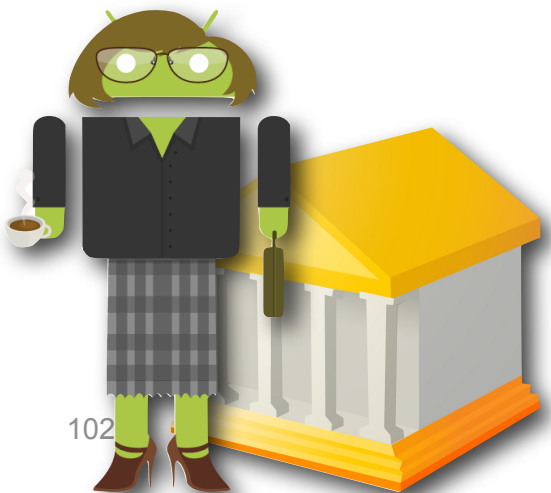
- OpenPGP key only needs to be uploaded once
- Once uploaded, we can make requests for encrypted records of user account activity and mailbox dumps

Audit API

Create and Check the Status of an Account Activity Report

Step 1: Make an Account Info Request

```
audit_service.createAccountInformationRequest (user='michael' )
```

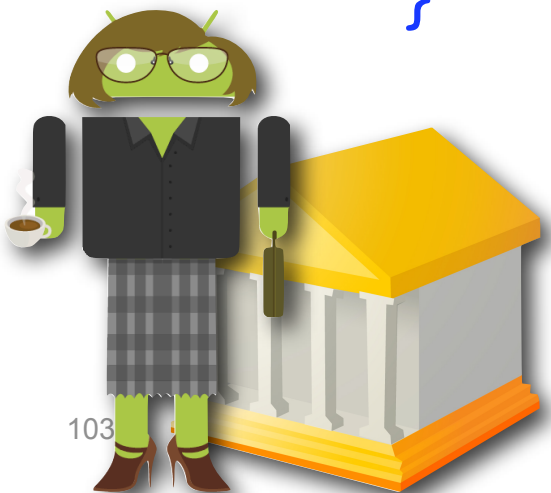


Audit API

Create and Check the Status of an Account Activity Report

Step 1: Make an Account Info Request

```
audit_service.createAccountInformationRequest(user='michael')  
  
{ 'status': 'PENDING',  
  'userEmailAddress': 'michael@yourdomain.com',  
  'adminEmailAddress': 'admin@yourdomain.com',  
  'requestDate': '2011-03-08 18:23',  
  'requestId': '19026018'  
}
```

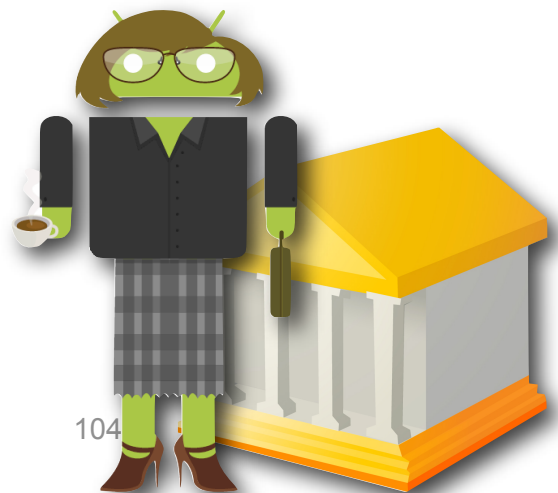


Audit API

Create and Check the Status of an Account Activity Report

Step 2: Make an Account Info Request

```
service.getAccountInformationRequestStatus(  
    user='michael',request_id='19026018')
```

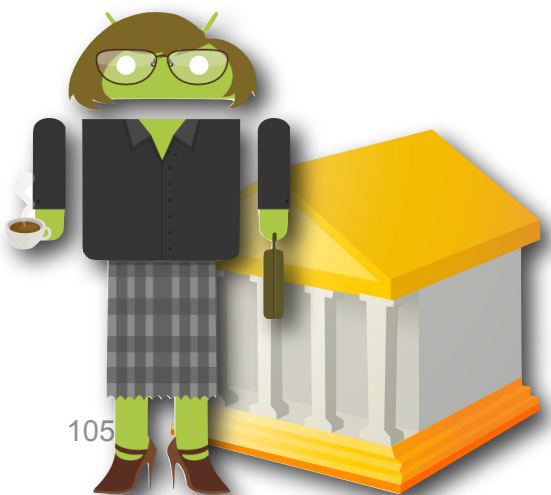


Audit API

Create and Check the Status of an Account Activity Report

Step 2: Make an Account Info Request

```
service.getAccountInformationRequestStatus(  
    user='michael',request_id='19026018')  
  
{  
    'status': 'PENDING',  
    'adminEmailAddress': 'admin@yourdomain.com',  
    'userEmailAddress': 'michael@yourdomain.com',  
    'requestDate': '2011-03-08 18:23', 'requestId': '19026018',  
}
```



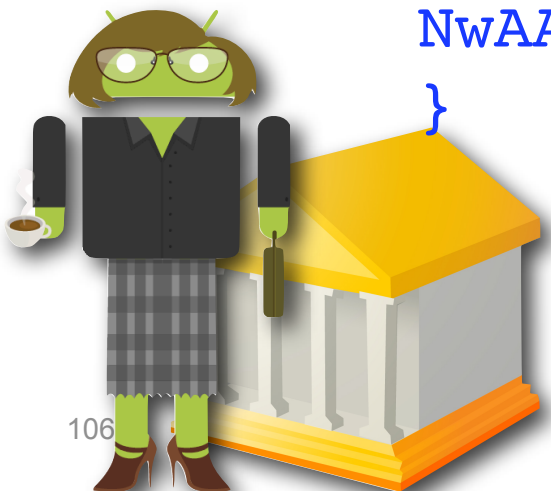
Audit API

Create and Check the Status of an Account Activity Report

Step 3: Download the Activity Report

```
service.getAccountInformationRequestStatus(  
    user='michael',request_id='19026018')
```

```
{'status': 'COMPLETED',  
  'adminEmailAddress': 'admin@yourdomain.com',  
  'userEmailAddress': 'michael@yourdomain.com',  
  'requestDate': '2011-03-08 18:23', 'requestId': '19026018',  
  'numberOfFiles': '1', 'completedDate': '2011-03-08 18:55',  
  'fileUrl0': 'https://apps-apis.google.com/a/data/compliance/audit/  
NwAAAJ54LYvzJW5uawJMEoFIU9zGbAy2IZBoz2YDc...'
```



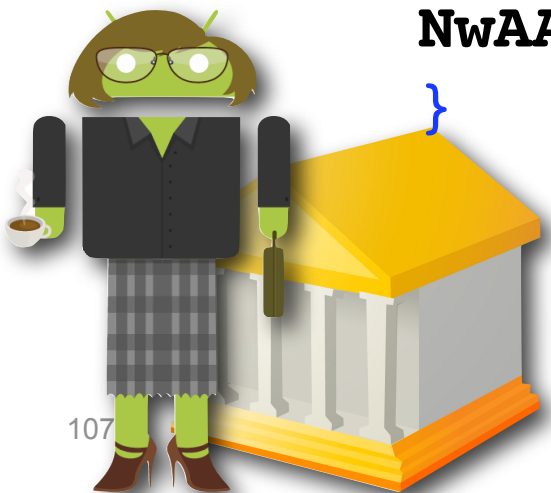
Audit API

Create and Check the Status of an Account Activity Report

Step 3: Download the Activity Report

```
service.getAccountInformationRequestStatus(  
    user='michael',request_id='19026018')
```

```
{  
  'status': 'COMPLETED',  
  'adminEmailAddress': 'admin@yourdomain.com',  
  'userEmailAddress': 'michael@yourdomain.com',  
  'requestDate': '2011-03-08 18:23', 'requestId': '19026018',  
  'numberOfFiles': '1', 'completedDate': '2011-03-08 18:55',  
  'fileUrl0': 'https://apps-apis.google.com/a/data/compliance/audit/  
NwAAAJ54LYvzJW5uawJMEoFIU9zGbAy2IZBoz2YDc...'  
}
```



Account Activity Report

<https://apps-apis.google.com/a/data/compliance/audit/NwAAAJ54LYvzJW5uawJMEoFIU9zGbAy2IZBoz2YDc...>



Account Activity Report

<https://apps-apis.google.com/a/data/compliance/audit/NwAAAJ54LYvzJW5uawJMEoFIU9zGbAy2IZBoz2YDc...>



Account Activity Report



<https://apps-apis.google.com/a/data/compliance/audit/NwAAAJ54LYvzJW5uawJMEoFIU9zGbAy2IZBoz2YDc...>

SubscriberInfo
Email michael@yourdomain.com
Status Enabled
Services Hosted Apps Accounts, Docs, Gmail, Calendar
Name Michael Manoochehri
Created on 2011-03-08 18:55Z
Lang en

Logs

All times are displayed in UTC/GMT.

michael@yourdomain.com

Date/Time Event IP

| | | | |
|------------|-----------|---------------|--------------|
| 2010-03-05 | 10:50:14Z | Login Success | 134.45.67.89 |
| 2010-03-05 | 09:40:14Z | Login Failure | 50.70.23.45 |
| 2010-03-05 | 08:20:14Z | Logout | 134.45.67.89 |
| 2010-03-05 | 08:00:14Z | Login Success | 134.45.67.89 |

Account Activity Report

SubscriberInfo

Email michael@yourdomain.com

Status Enabled

Services Hosted Apps Accounts, Docs, Gmail, Calendar

Name Michael Manoochehri

Created on 2011-03-08 18:55Z

Lang en

Logs

All times are displayed in UTC/GMT.

michael@yourdomain.com

Date/Time Event IP

2010-03-05 10:50:14Z Login Success 134.45.67.89

2010-03-05 09:40:14Z Login Failure 50.70.23.45

2010-03-05 08:20:14Z Logout 134.45.67.89

2010-03-05 08:00:14Z Login Success 134.45.67.89

Account Activity Report

SubscriberInfo

Email michael@yourdomain.com

Status Enabled

Services Hosted Apps Accounts, Docs, Gmail, Calendar

Name Michael Manoochehri

Created on 2011-03-08 18:55Z

Lang en

Logs

All times are displayed in UTC/GMT.

michael@yourdomain.com

Date/Time Event IP

2010-03-05 10:50:14Z Login Success 134.45.67.89

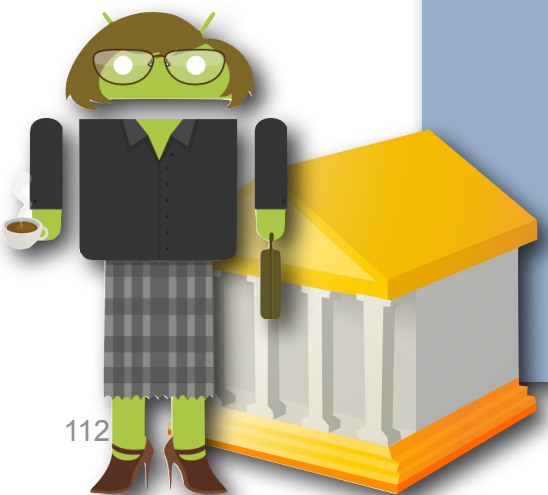
2010-03-05 09:40:14Z Login Failure 50.70.23.45

2010-03-05 08:20:14Z Logout 134.45.67.89

2010-03-05 08:00:14Z Login Success 134.45.67.89

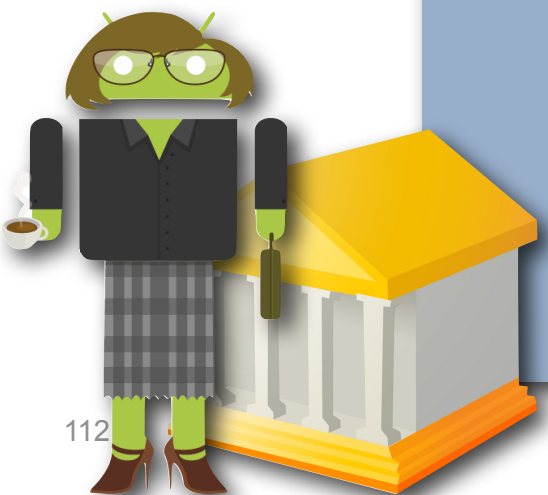
Audit API

Account Activity Report



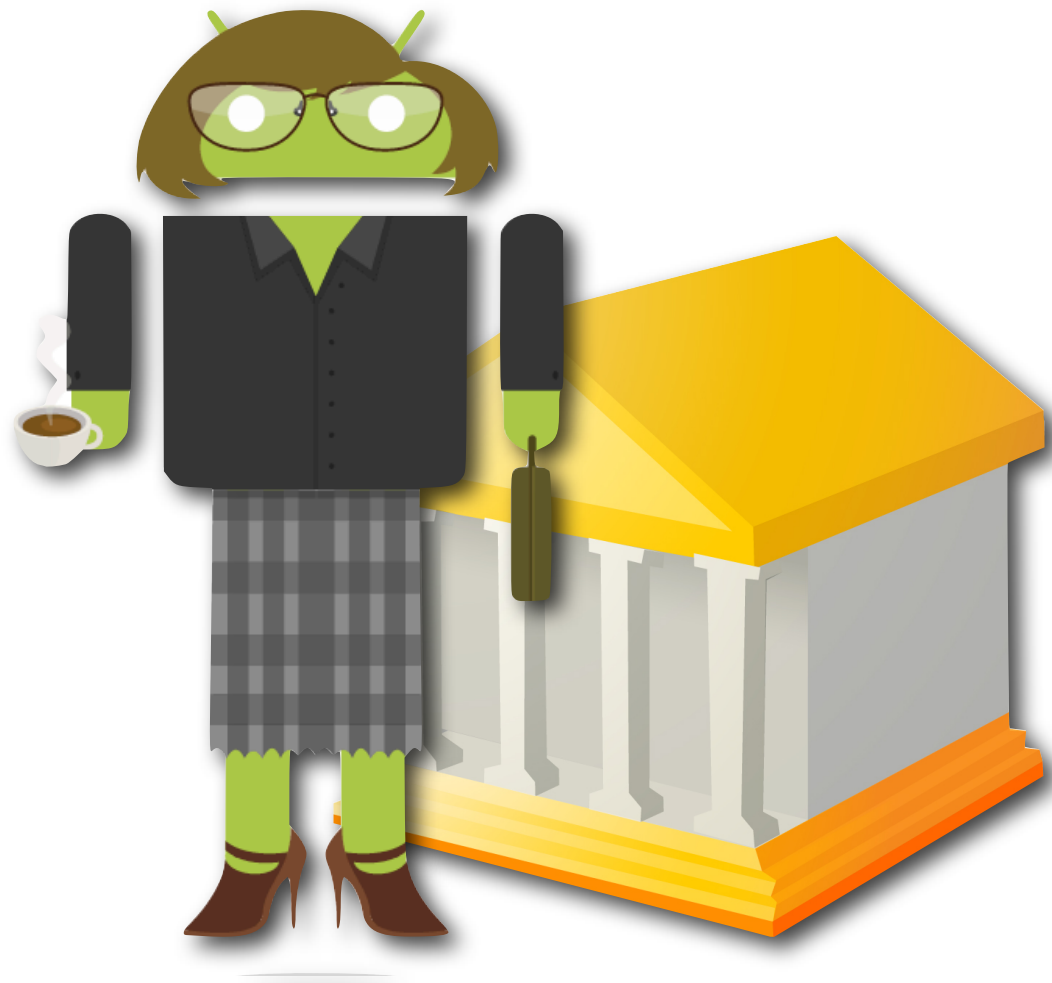
Audit API

Account Activity Report



2-Factor Authentication

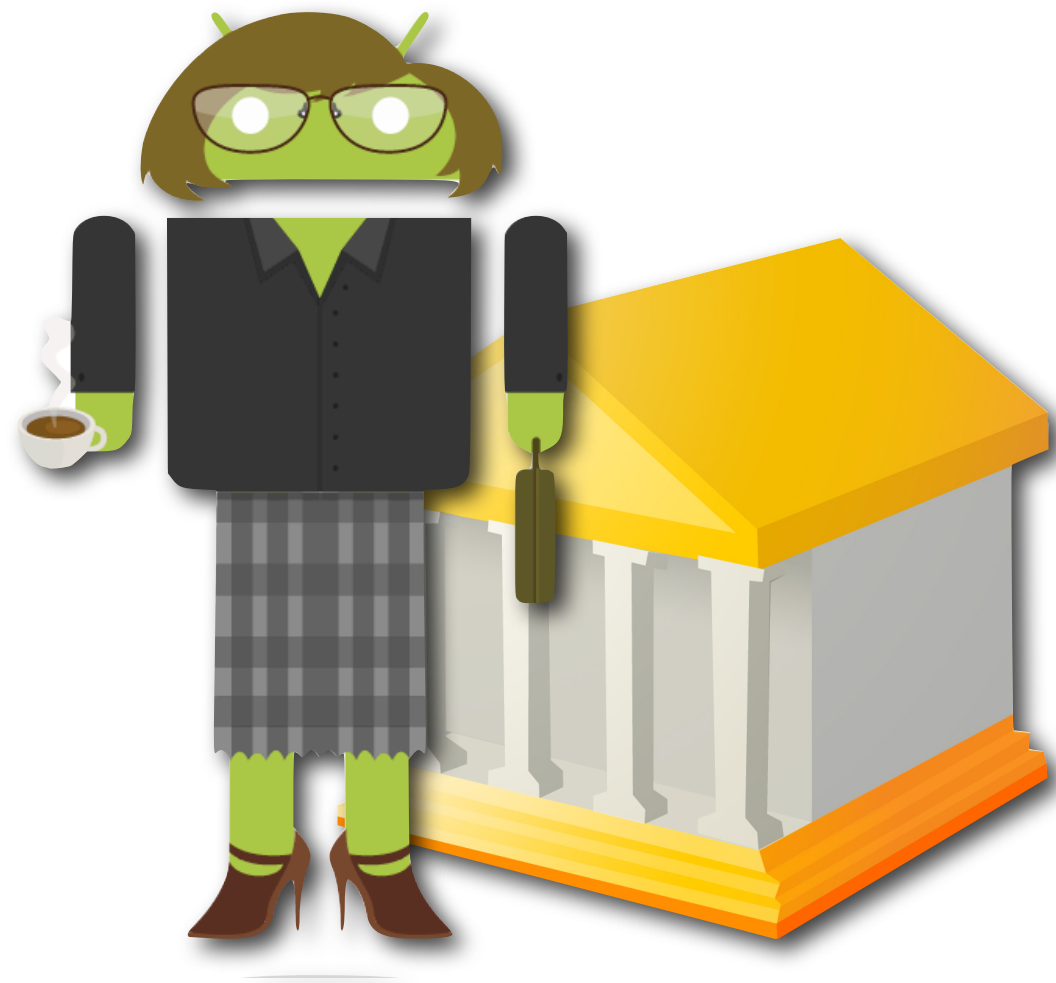
Something You Know



2-Factor Authentication

Something You Know

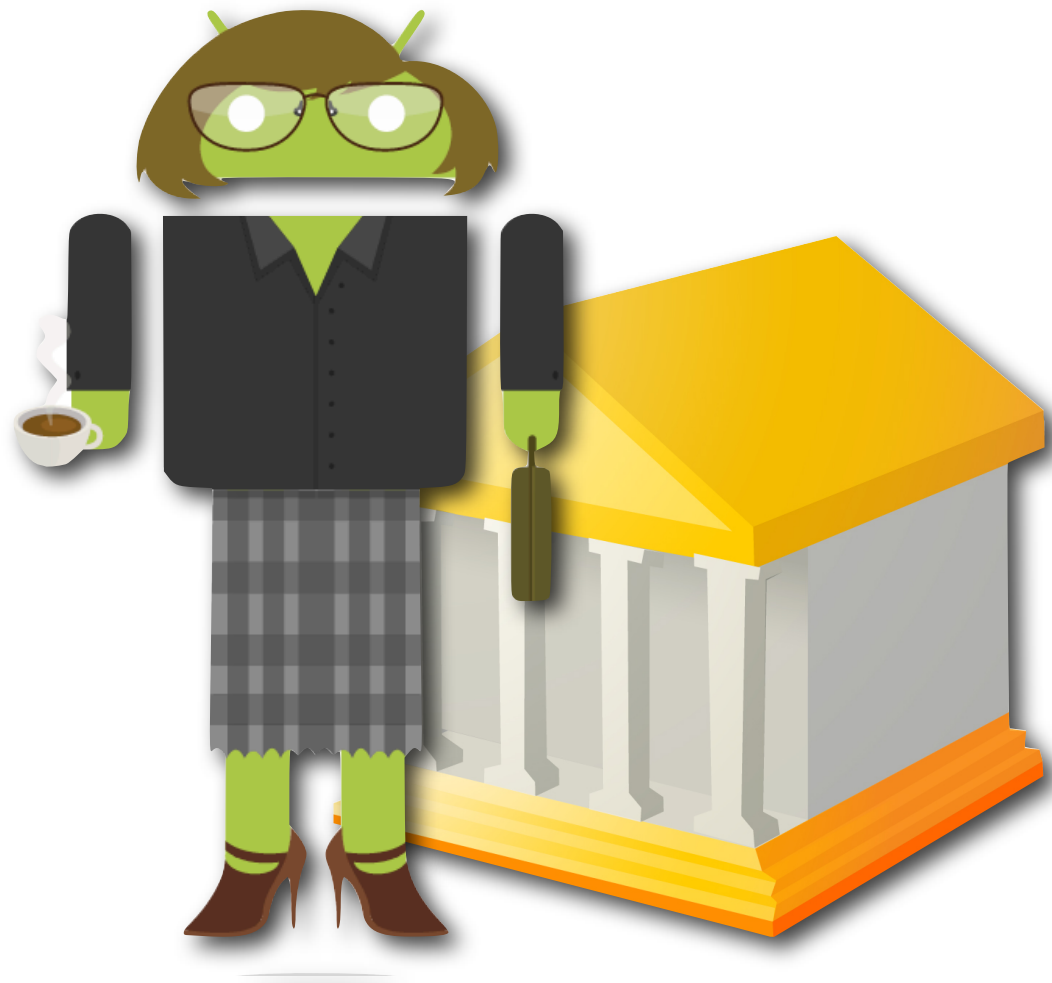
p@s5W0rd



2-Factor Authentication

Something You Know

p@s5W0rd



Something You Have



2-Factor Authentication

2-Factor Authentication

Sign in with your
Google Account

Username:
ex: pat@example.com

Password:

Stay signed in

[Can't access your account?](#)

2-Factor Authentication

Sign in with your
Google Account

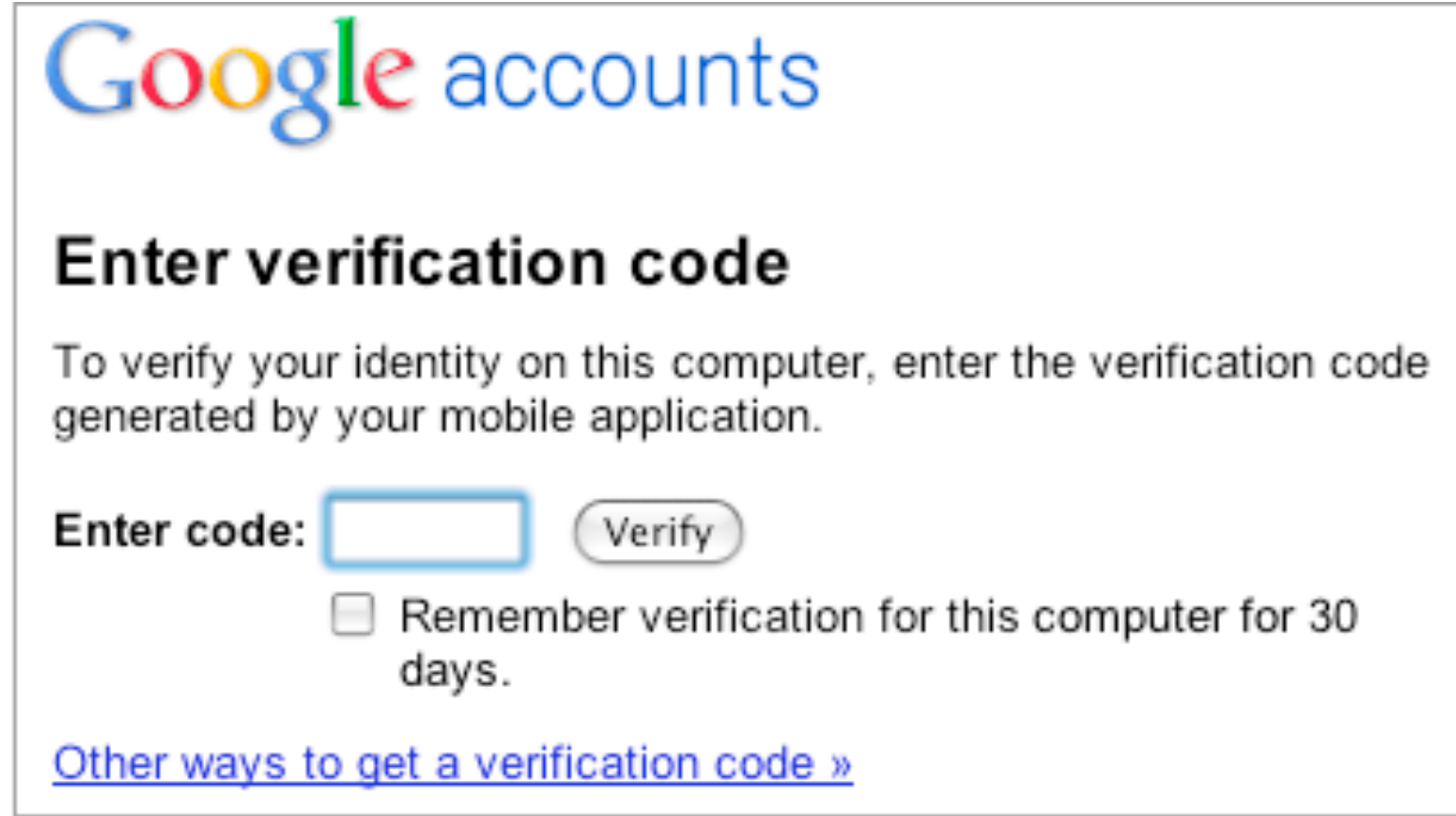
Username:
ex: pat@example.com

Password:

Stay signed in

[Can't access your account?](#)

2-Factor Authentication

A screenshot of the Google accounts verification interface. At the top left is the 'Google accounts' logo. Below it is the heading 'Enter verification code'. A paragraph of text explains that the user must enter a code from a mobile application to verify their identity on the computer. The form includes a label 'Enter code:' followed by a text input field and a 'Verify' button. Below the input field is a checkbox labeled 'Remember verification for this computer for 30 days.' At the bottom, there is a blue link that says 'Other ways to get a verification code »'.

Google accounts

Enter verification code

To verify your identity on this computer, enter the verification code generated by your mobile application.

Enter code:

Remember verification for this computer for 30 days.

[Other ways to get a verification code »](#)

2-Factor Authentication



Google accounts

Enter verification code

To verify your identity on this computer, enter the verification code generated by your mobile application.

Enter code:

Remember verification for this computer for 30 days.

[Other ways to get a verification code »](#)

2-Factor Auth: Application-Specific Passwords

For applications that lack second factor input access: POP/IMAP clients, etc.



2-Factor Auth: Application-Specific Passwords

For applications that lack second factor input access: POP/IMAP clients, etc.



Application-specific password generated

You may now enter your new application-specific password into your application. For security reasons, it will not be displayed again:

bbtb g3tb 5bp7 4ocj
Spaces don't matter.
You should need to enter this password only once - no need to memorize it.

[Hide password](#)

| Your application-specific passwords | Creation date | |
|-------------------------------------|---------------|----------------------------|
| Android Mail | Feb 8, 2011 | [Revoke] |
| Outlook | Feb 8, 2011 | [Revoke] |
| AdWords Editor - Desktop | Feb 10, 2011 | [Revoke] |

Compliance and Security Tools



Audit API

Legal Compliance API

2-Factor Authentication

Stronger Google Apps Authentication Flow

Google Apps Administrative API Toolbox



Provisioning API

Create and Update Users, Groups, Nicknames, OrgUnits



Email Migration API

Move your Legacy Email to the Cloud



Reporting API

Generate Reports about User Statistics



Admin & Email Settings APIs

The Remote Control for Administrative Settings

Google Apps Administrative API Toolbox



Provisioning API

Create and Update Users, Groups, Nicknames, OrgUnits



Email Migration API

Move your Legacy Email to the Cloud



Reporting API

Generate Reports about User Statistics



Admin & Email Settings APIs

The Remote Control for Administrative Settings



Audit API

Conduct Audits on User Activity and Data

How Google Apps Adds Value



Security and Availability

Same uptime and infrastructure used for Google products



Built-in Enterprise Security Features

2-Factor Authentication, Single Sign On, Reporting Tools

How Google Apps Adds Value



Security and Availability

Same uptime and infrastructure used for Google products



Built-in Enterprise Security Features

2-Factor Authentication, Single Sign On, Reporting Tools



A Toolbox of Administrative APIs

Reporting, Compliance, Identity Management and more...

Next steps...

- Documentation about APIs used in this session:
<http://code.google.com/googleapps/docs>
- Post feedback about this session:
<http://goo.gl/zCT1M>
- Stay connected! Read the Google Apps Developer Blog
<http://googleappsdeveloper.blogspot.com>
- Join our Admin API Developer Google Group:
<http://goo.gl/V55TD>

Q & A

Google™

