CAN YOU KEEP A SECRET?

@AARONBASSETT

nexmo®
The Vonage® API Platform

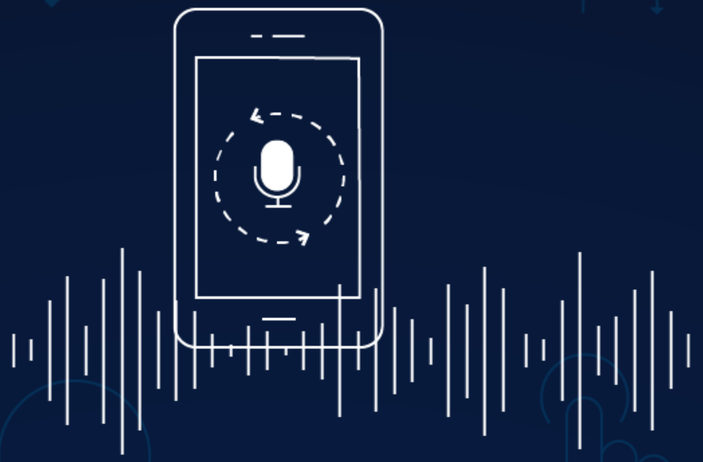
AARON BASSETT

DEVELOPER ADVOCATE

nexmo®
The Vonage® API Platform

nexmo®

The Vonage® API Platform



```
import nexmo

client = nexmo.Client(
    key="a925db1ar392", secret="01nd637fn29oe31mc721"
)
client.send_message({
    "from": "Python",
    "to": "447700900981",
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import nexmo

client = nexmo.Client(
    key="a925db1ar392", secret="01nd637fn29oe31mc721"
)
client.send_message({
    "from": "Python",
    "to": "447700900981",
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import nexmo

NEXMO_KEY = "a925db1ar392"
NEXMO_SECRET = "01nd637fn29oe31mc721"
MY_NUMBER = "447700900981"

client = nexmo.Client(key=NEXMO_KEY, secret=NEXMO_SECRET)
client.send_message({
    "from": "Python",
    "to": MY_NUMBER,
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import nexmo

NEXMO_KEY = "a925db1ar392"
NEXMO_SECRET = "01nd637fn29oe31mc721"
MY_NUMBER = "447700900981"

client = nexmo.Client(key=NEXMO_KEY, secret=NEXMO_SECRET)
client.send_message({
    "from": "Python",
    "to": MY_NUMBER,
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import nexmo

NEXMO_KEY = "a925db1ar392"
NEXMO_SECRET = "01nd637fn29oe31mc721"
MY_NUMBER = "447700900981"

client = nexmo.Client(key=NEXMO_KEY, secret=NEXMO_SECRET)
client.send_message({
    "from": "Python",
    "to": MY_NUMBER,
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```
git add .
git commit -m “wip”
git push
```

nexmo®

The Vonage® API Platform

"THE MEDIAN TIME TO DISCOVERY WAS 20 SECONDS, WITH TIMES RANGING FROM HALF A SECOND TO OVER 4 MINUTES"

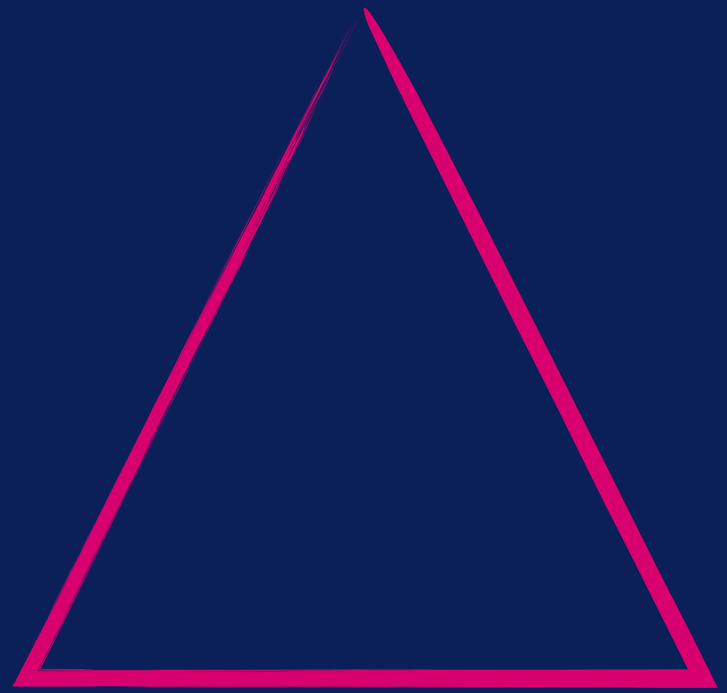
– How Bad Can It Git? Characterizing Secret Leakage in Public GitHub Repositories

NOT SAFE, BUT VERY EASY

nexmo®

The Vonage® API Platform

FUNCTIONALITY



SECURITY

USABILITY

nexmo®

The Vonage® API Platform

LOW FRICTION

nexmo®

The Vonage® API Platform

EASY TO IMPLEMENT

nexmo®

The Vonage® API Platform

12 FACTOR APPS

nexmo®

The Vonage® API Platform

I. CODEBASE

II. DEPENDENCIES

III. CONFIG

IV. BACKING SERVICES

V. BUILD, RELEASE, RUN

VI. PROCESSES

VII. PORT BINDING

VIII. CONCURRENCY

IX. DISPOSABILITY

X. DEV/PROD PARITY

XI. LOGS

XII. ADMIN PROCESSES

III. CONFIG

nexmo®

The Vonage® API Platform

"A litmus test for whether an app has all config correctly factored out of the code is whether the codebase could be made open source at any moment, without compromising any credentials."

—The twelve-factor app

ENVIRONMENT VARIABLES

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.environ["NEXMO_KEY"], secret=os.environ["NEXMO_SECRET"]
)

client.send_message({
    "from": "Python",
    "to": os.environ["MY_NUMBER"],
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.environ["NEXMO_KEY"], secret=os.environ["NEXMO_SECRET"]
)

client.send_message({
    "from": "Python",
    "to": os.environ["MY_NUMBER"],
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.environ.get("NEXMO_KEY"),
    secret=os.environ.get("NEXMO_SECRET")
)

client.send_message({
    "from": "Python",
    "to": os.environ.get("MY_NUMBER"),
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.environ.get("NEXMO_KEY"),
    secret=os.environ.get("NEXMO_SECRET")
)

client.send_message({
    "from": "Python",
    "to": os.environ.get("MY_NUMBER"),
    "text": "Hello world"
})
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.getenv("NEXMO_KEY"),
    secret=os.getenv("NEXMO_SECRET"),
)
client.send_message(
    {
        "from": "Python",
        "to": os.getenv("MY_NUMBER"),
        "text": "Hello world",
    }
)
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo

client = nexmo.Client(
    key=os.getenv("NEXMO_KEY"),
    secret=os.getenv("NEXMO_SECRET"),
)
client.send_message(
    {
        "from": "Python",
        "to": os.getenv("MY_NUMBER"),
        "text": "Hello world",
    }
)
```

nexmo®

The Vonage® API Platform

CREATING ENVIRONMENT VARIABLES

nexmo®

The Vonage® API Platform

```
# This file must be used with "source bin/activate" *from bash*
# you cannot run it directly

deactivate () {
    ...

    # Unset variables
    unset NEXMO_KEY
    unset NEXMO_SECRET
    unset MY_NUMBER
}

...

export NEXMO_KEY="a925db1ar392"
export NEXMO_SECRET="01nd637fn29oe31mc721"
export MY_NUMBER="447700900981"
```

nexmo®
The Vonage® API Platform

```
# This file must be used with "source bin/activate" *from bash*
# you cannot run it directly

deactivate () {
    ...

    # Unset variables
    unset NEXMO_KEY
    unset NEXMO_SECRET
    unset MY_NUMBER
}

...

export NEXMO_KEY="a925db1ar392"
export NEXMO_SECRET="01nd637fn29oe31mc721"
export MY_NUMBER="447700900981"
```

```
# This file must be used with "source bin/activate" *from bash*
# you cannot run it directly

deactivate () {
    ...

    # Unset variables
    unset NEXMO_KEY
    unset NEXMO_SECRET
    unset MY_NUMBER
}

...

export NEXMO_KEY="a925db1ar392"
export NEXMO_SECRET="01nd637fn29oe31mc721"
export MY_NUMBER="447700900981"
```

"direnv is an extension for your shell. It augments existing shells with a new feature that can load and unload environment variables depending on the current directory."

– direnv.net

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
```

nexmo®

The Vonage® API Platform

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
.envrc is not allowed
```

nexmo®

The Vonage® API Platform

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
.envrc is not allowed
$ direnv allow .
```

nexmo®
The Vonage® API Platform

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
.envrc is not allowed
$ direnv allow .
direnv: reloading
direnv: loading .envrc
direnv export: +NEXMO_KEY
```

nexmo®

The Vonage® API Platform

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
.envrc is not allowed
$ direnv allow .
direnv: reloading
direnv: loading .envrc
direnv export: +NEXMO_KEY
$ cd ..
```

nexmo®

The Vonage® API Platform

```
$ echo export NEXMO_KEY=a925db1ar392 > .envrc
.envrc is not allowed
$ direnv allow .
direnv: reloading
direnv: loading .envrc
direnv export: +NEXMO_KEY
$ cd ..
direnv: unloading
```

nexmo®

The Vonage® API Platform

IGNORE .ENVRC

```
echo .envrc > ~/.gitignore
git config --global core.excludesfile ~/.gitignore
```

nexmo®
The Vonage® API Platform

EXAMPLE .ENVRC

```
grep -ohr "^export .*=" .envrc > .envrc.example
```

nexmo®
The Vonage® API Platform

SHARING VALUES & FILES

nexmo®

The Vonage® API Platform

```python
import os
import nexmo


client = nexmo.Client(
    application_id=os.getenv("NEXMO_APPLICATION_ID"),
    private_key=os.getenv("NEXMO_PRIVATE_KEY", "./private.key"),
)

response = client.create_call(
    {
        "to": [{"type": "phone", "number": os.getenv("TO_NUMBER")}],
        "from": {"type": "phone", "number": os.getenv("NEXMO_NUMBER")},
        "answer_url": [f"{os.getenv('VAPI_URL')}/answer"],
    }
)
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo


client = nexmo.Client(
    application_id=os.getenv("NEXMO_APPLICATION_ID"),
    private_key=os.getenv("NEXMO_PRIVATE_KEY", "./private.key"),
)

response = client.create_call(
    {
        "to": [{"type": "phone", "number": os.getenv("TO_NUMBER")}],
        "from": {"type": "phone", "number": os.getenv("NEXMO_NUMBER")},
        "answer_url": [f"{os.getenv('VAPI_URL')}/answer"],
    }
)
```

nexmo®

The Vonage® API Platform

```python
import os
import nexmo


client = nexmo.Client(
    application_id=os.getenv("NEXMO_APPLICATION_ID"),
    private_key=os.getenv("NEXMO_PRIVATE_KEY", "./private.key"),
)

response = client.create_call(
    {
        "to": [{"type": "phone", "number": os.getenv("TO_NUMBER")}],
        "from": {"type": "phone", "number": os.getenv("NEXMO_NUMBER")},
        "answer_url": [f"{os.getenv('VAPI_URL')}/answer"],
    }
)
```

nexmo®

The Vonage® API Platform

git-secret

GIT-SECRET.IO

nexmo®

The Vonage® API Platform

"git-secret encrypts files and stores them inside the git repository. So you will have all the changes for every commit."

—git-secret.io

SAFE & EASY

nexmo®

The Vonage® API Platform

SAFE & EASY-ISH

nexmo®

The Vonage® API Platform

PGP

nexmo®

The Vonage® API Platform

```
$ git secret init
git-secret: init created: '/myproject/.gitsecret/'
```

```
$ git secret tell me@aaronbassett.com
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   1  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2021-09-30
gpg: keybox '/myproject/.gitsecret/keys/pubring.kbx' created
gpg: /myproject/.gitsecret/keys/trustdb.gpg: trustdb created
git-secret: done. me@aaronbassett.com added as user(s) who know the secret.
```

```
$ git secret add private.key
git-secret: these files are not in .gitignore: private.key
git-secret: auto adding them to .gitignore
git-secret: 1 item(s) added.
```

```
$ cat .gitignore
.gitsecret/keys/random_seed
!*.secret
private.key
```

nexmo®

The Vonage® API Platform

```
$ ls
.git
.gitignore
.gitsecret
private.key
```

nexmo®

The Vonage® API Platform

```
$ git secret hide
git-secret: done. 1 of 1 files are hidden.
```

nexmo®

The Vonage® API Platform

```
$ ls
.git
.gitignore
.gitsecret
private.key
private.key.secret
```

nexmo®

The Vonage® API Platform

```
$ git secret reveal
File '/myproject/private.key' exists. Overwrite? (y/N) y
git-secret: done. 1 of 1 files are revealed.
```

nexmo®
The Vonage® API Platform

```
$ git secret whoknows
me@aaronbassett.com

$ git secret list
private.key
```

nexmo®
The Vonage® API Platform

```
$ git secret killperson me@aaronbassett.com
git-secret: removed keys.
git-secret: now [me@aaronbassett.com] do not have an access to the repository.
git-secret: make sure to hide the existing secrets again.

$ git secret reveal
git-secret: abort: no public keys for users found. run 'git secret tell email@address.
```

GIT SECRETS

nexmo®

The Vonage® API Platform

"git-secrets scans commits, commit messages, and --no-ff merges to prevent adding secrets into your git repositories. If a commit, commit message, or any commit in a --no-ff merge history matches one of your configured prohibited regular expression patterns, then the commit is rejected."

– awslabs/git-secrets

nexmo®
The Vonage® API Platform

```
$ git secrets --register-aws --global
OK

$ git secrets --install ~/.git-templates/git-secrets
√ Installed commit-msg hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/commit-msg
√ Installed pre-commit hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/pre-commit
√ Installed prepare-commit-msg hook to /Users/aaronbassett/.git-templates/
git-secrets/hooks/prepare-commit-msg

$ git config --global init.templateDir ~/.git-templates/git-secrets
```

```
$ git secrets --register-aws --global
OK

$ git secrets --install ~/.git-templates/git-secrets
✓ Installed commit-msg hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/commit-msg
✓ Installed pre-commit hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/pre-commit
✓ Installed prepare-commit-msg hook to /Users/aaronbassett/.git-templates/
git-secrets/hooks/prepare-commit-msg

$ git config --global init.templateDir ~/.git-templates/git-secrets
```

```
$ git secrets --register-aws --global
OK

$ git secrets --install ~/.git-templates/git-secrets
√ Installed commit-msg hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/commit-msg
√ Installed pre-commit hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/pre-commit
√ Installed prepare-commit-msg hook to /Users/aaronbassett/.git-templates/
git-secrets/hooks/prepare-commit-msg

$ git config --global init.templateDir ~/.git-templates/git-secrets
```

```
$ git secrets --register-aws --global
OK

$ git secrets --install ~/.git-templates/git-secrets
√ Installed commit-msg hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/commit-msg
√ Installed pre-commit hook to /Users/aaronbassett/.git-templates/git-
secrets/hooks/pre-commit
√ Installed prepare-commit-msg hook to /Users/aaronbassett/.git-templates/
git-secrets/hooks/prepare-commit-msg

$ git config --global init.templateDir ~/.git-templates/git-secrets
```

PROVIDERS

nexmo®

The Vonage® API Platform

```
^[5KL][1-9A-HJ-NP-Za-km-z]{50,51}$
(xox[p|b|o|a]-[0-9]{12}-[0-9]{12}-[0-9]{12}-[a-z0-9]{32})
https://hooks.slack.com/services/T[a-zA-Z0-9_]{8}/B[a-zA-Z0-9_]{8}/[a-zA-Z0-9_]{24}
EAACEdEose0cBA[0-9A-Za-z]+
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*[1-9][0-9]+-[0-9a-zA-Z]{40}
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*['|\"][0-9a-zA-Z]{35,44}['|\"]
AIza[0-9A-Za-z\\-_]{35}
[0-9]+-[0-9A-Za-z_]{32}\\.apps\\.googleusercontent\\.com
ya29\\.[0-9A-Za-z\\-_]+
[h|H][e|E][r|R][o|O][k|K][u|U].*[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}
(-----(BEGIN|END) PRIVATE KEY-----)
(-----(BEGIN|END) RSA PRIVATE KEY-----)
```

```
$ git secrets --add-provider -- cat /secret/file/patterns
```

```
(-----(BEGIN|END) RSA PRIVATE KEY-----)
```

nexmo®

The Vonage® API Platform

```
ya29\\.[0-9A-Za-z\\-_]+
```

nexmo®

The Vonage® API Platform

```
EAACEdEose0cBA[0-9A-Za-z]+
```

nexmo®

The Vonage® API Platform

```
$ cd /secret/file/patterns
$ ls
crypto
keys
vendors
```

nexmo®

The Vonage® API Platform

```
# Slack
(xox[p|b|o|a]-[0-9]{12}-[0-9]{12}-[0-9]{12}-[a-z0-9]{32})
https://hooks.slack.com/services/T[a-zA-Z0-9_]{8}/B[a-zA-Z0-9_]{8}/[a-zA-Z0-9_]{24}

# Facebook
EAACEdEose0cBA[0-9A-Za-z]+

# Twitter
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*[1-9][0-9]+-[0-9a-zA-Z]{40}
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*['|\"][0-9a-zA-Z]{35,44}['|\"]

# Google
AIza[0-9A-Za-z\\-_]{35}
[0-9]+-[0-9A-Za-z_]{32}\\.apps\\.googleusercontent\\.com
ya29\\.[0-9A-Za-z\\-_]+

# Heroku
[h|H][e|E][r|R][o|O][k|K][u|U].*[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}
```

```
git secrets --add-provider -- egrep -rhv "(^#|^$)" /secret/file/patterns
```

nexmo®

The Vonage® API Platform

```
^[5KL][1-9A-HJ-NP-Za-km-z]{50,51}$
(xox[p|b|o|a]-[0-9]{12}-[0-9]{12}-[0-9]{12}-[a-z0-9]{32})
https://hooks.slack.com/services/T[a-zA-Z0-9_]{8}/B[a-zA-Z0-9_]{8}/[a-zA-Z0-9_]{24}
EAACEdEose0cBA[0-9A-Za-z]+
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*[1-9][0-9]+-[0-9a-zA-Z]{40}
[t|T][w|W][i|I][t|T][t|T][e|E][r|R].*['|\"][0-9a-zA-Z]{35,44}['|\"]
AIza[0-9A-Za-z\\-_]{35}
[0-9]+-[0-9A-Za-z_]{32}\\.apps\\.googleusercontent\\.com
ya29\\.[0-9A-Za-z\\-_]+
[h|H][e|E][r|R][o|O][k|K][u|U].*[0-9A-F]{8}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{4}-[0-9A-F]{12}
(-----(BEGIN|END) PRIVATE KEY-----)
(-----(BEGIN|END) RSA PRIVATE KEY-----)
```

```
$ git add ‘private.key’
$ git commit -m "Adding some files I shouldn't"
private.key:1:-----BEGIN PRIVATE KEY-----

[ERROR] Matched one or more prohibited patterns

Possible mitigations:
- Mark false positives as allowed using: git config --add secrets.allowed ...
- Mark false positives as allowed by adding regular expressions to .gitallowed at repository's root directory
- List your configured patterns: git config --get-all secrets.patterns
- List your configured allowed patterns: git config --get-all secrets.allowed
- List your configured allowed patterns in .gitallowed at repository's root directory
- Use --no-verify if this is a one-time false positive
```

GITLEAKS

nexmo®

The Vonage® API Platform

"AUDIT GIT REPOS FOR SECRETS. GITLEAKS PROVIDES A WAY FOR YOU TO FIND UNENCRYPTED SECRETS AND OTHER UNWANTED DATA TYPES IN GIT SOURCE CODE REPOSITORIES"

— Zri cEthEZaV/9I tLEaKS

nexmo®

The Vonage® API Platform

I. A GIT REPO

II. GITHUB USER

III. GITHUB ORGANIZATION

IV. GITHUB PR

V. GITLAB USER

VI. GITLAB GROUP

```
{
    "line": "-----BEGIN PRIVATE KEY-----",
    "commit": "37f19780583f12fd2fb687f2d7d3840880e79c76",
    "offender": "-----BEGIN PRIVATE KEY-----",
    "rule": "PKCS8",
    "info": "-----BEGIN PRIVATE KEY----- regex match",
    "commitMsg": "wip backup ",
    "author": "Joe Bloggs",
    "email": "jbloggs@example.com",
    "file": "app/private.key",
    "repo": "my-awesome-app",
    "date": "2019-09-14T12:26:10+08:00",
    "tags": "key, PKCS8",
    "severity": ""
  }
```

```
gitleaks —github-org=MyOrg > /dev/null
if [ $? -eq 0 ]; then
    echo "No Leaks"
else
    nexmo sms 447700900235 Git is leaking --confirm
fi
```

nexmo®

The Vonage® API Platform

```
gitleaks —github-org=MyOrg > /dev/null
if [ $? -eq 0 ]; then
    echo "No Leaks"
else
    nexmo sms 447700900235 Git is leaking --confirm
fi
```

nexmo®

The Vonage® API Platform

```
gitleaks —github-org=MyOrg > /dev/null
if [ $? -eq 0 ]; then
    echo "No Leaks"
else
    nexmo sms 447700900235 Git is leaking --confirm
fi
```

nexmo®

The Vonage® API Platform

```
gitleaks —github-org=MyOrg > /dev/null
if [ $? -eq 0 ]; then
    echo "No Leaks"
else
    nexmo sms 447700900235 Git is leaking --confirm
fi
```

nexmo®

The Vonage® API Platform

1. KEEP SECRETS AND CODE SEPARATE

nexmo®
The Vonage® API Platform

2. IF YOU NEED TO SHARE SECRETS ENCRYPT THEM FIRST

PGP IS A PITA, USE TOOLS TO MAKE IT EASIER

3. AUTOMATE, AUTOMATE, AUTOMATE

nexmo®

The Vonage® API Platform

4. LATE IS BETTER THAN NEVER

nexmo®
The Vonage® API Platform

@AARONBASSETT
NOTI.ST/AARONBASSETT

nexmo®
The Vonage® API Platform