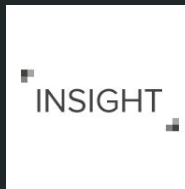


Bits on Bitcoin

a twist on verifying software releases

Justin Moore



In 2019 cybersecurity industry expenditures are at all time highs.

I ask myself: are users safer?

Security doesn't appear to scale with spending.

Hackers dropped a secret backdoor in Asus' update software

Zack Whittaker @zackwhittaker · 2 weeks ago

Comment



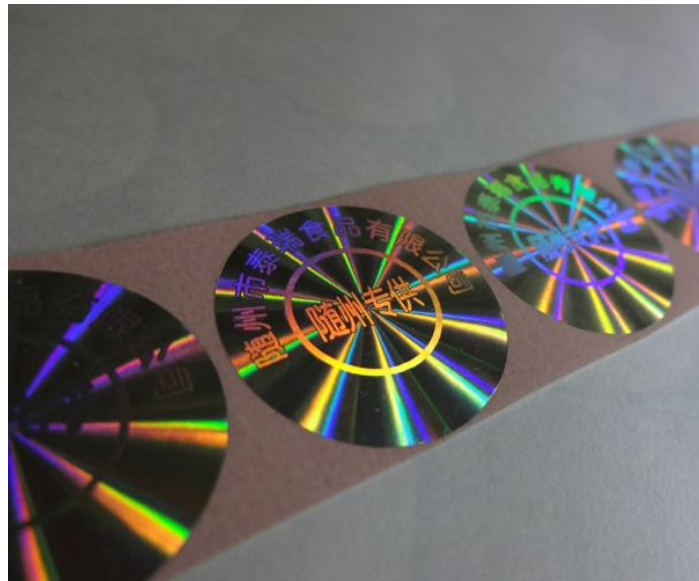
March 25, 2019

ASUS®

“The hackers took a real update from 2015 and subtly modified it before pushing it out to Asus customers...”

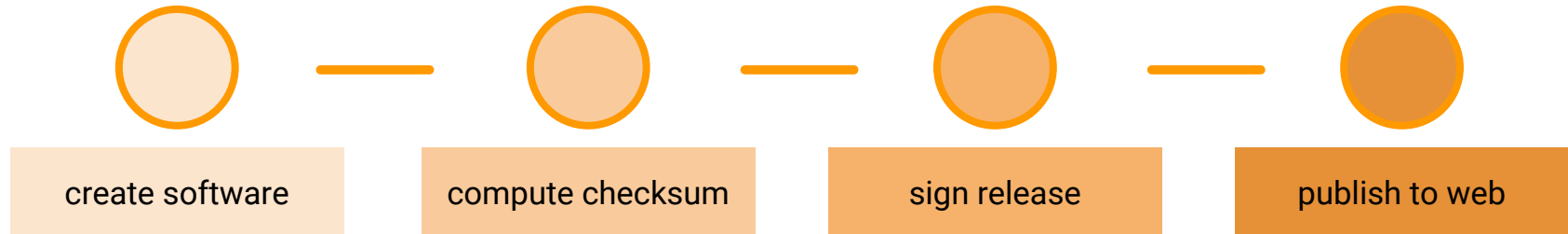
What do I mean by data security?

- 1) Integrity (checksums)
- 2) Authenticity (digital signatures)

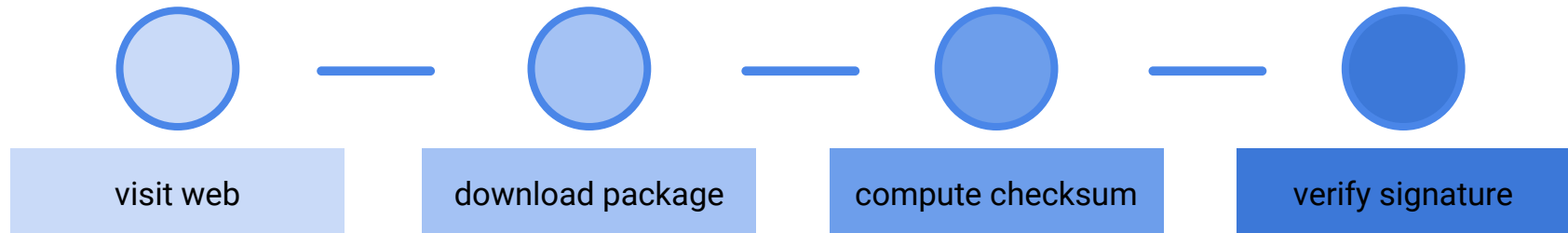


Software release/install process

Developer Publishing Software



End User Installing Software



There's nothing wrong with this approach.

...except that hardly any users do it.



I want to suggest a new foundation.

Addresses are information vehicles.

Software
Checksum



Address

Vanity-Mined
Developer Public
Key Hash



Address

1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG
DCSV - Decentralized Consensus Silicon Valley

Software
Checksum



Address

What's going to happen:

INSIGHT Vanity_Address 1936

Block #571154

DATE

PAY TO THE ORDER OF Checksum_Address \$ < \$0.10

Zero and less than 10/100 DOLLARS

Security Features Details on back

Bits on Bitcoin FOR Valid TX == Valid Signature

⑆0000000186⑆ 0000000529⑈ 1000

| | |
|---------------|--|
| Vanity_Addr | 1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG |
| Checksum_Addr | 1P1niSKDvNw7VCAA5FSywS95fDF9XMBgHH |
| Transaction | 5b42ea02de5a7d42336d97794c0c86c79540a5a768eccd5d7ee7b9a217bece88 |

Demo

Future work

Developer

Add multisig to the “FROM” address for additional security.

E.g. 3 of 5 developers need to sign

End User

Tweak installer such that shasum and tx verification happen automatically

User is unaware they're being secured.



github.com/humanumbrella/bitcoin-chksm

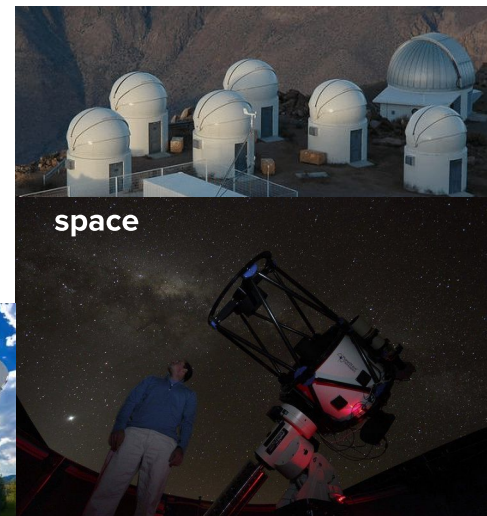
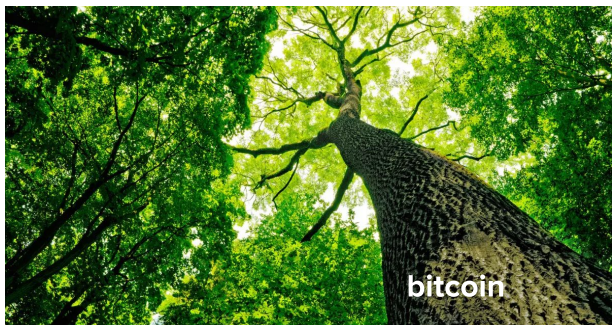
Justin P. Moore



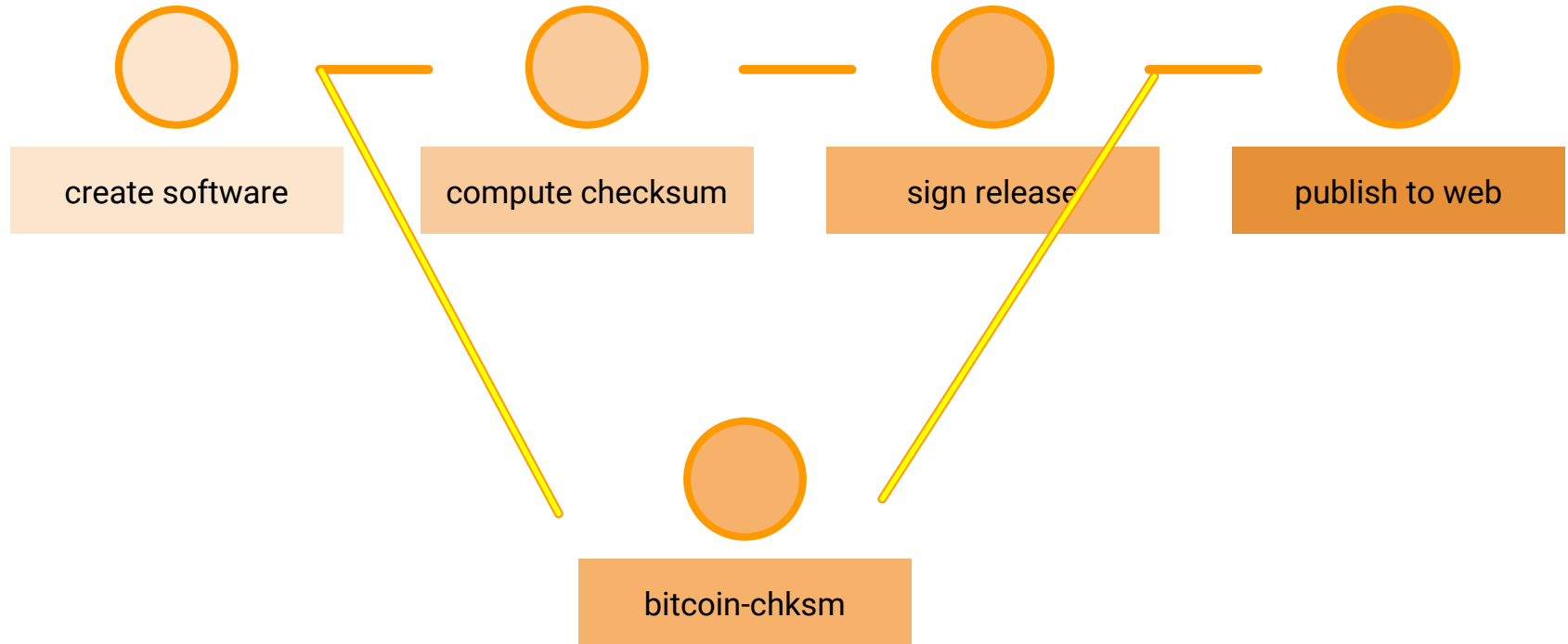
B.S. Computer Science and Physics



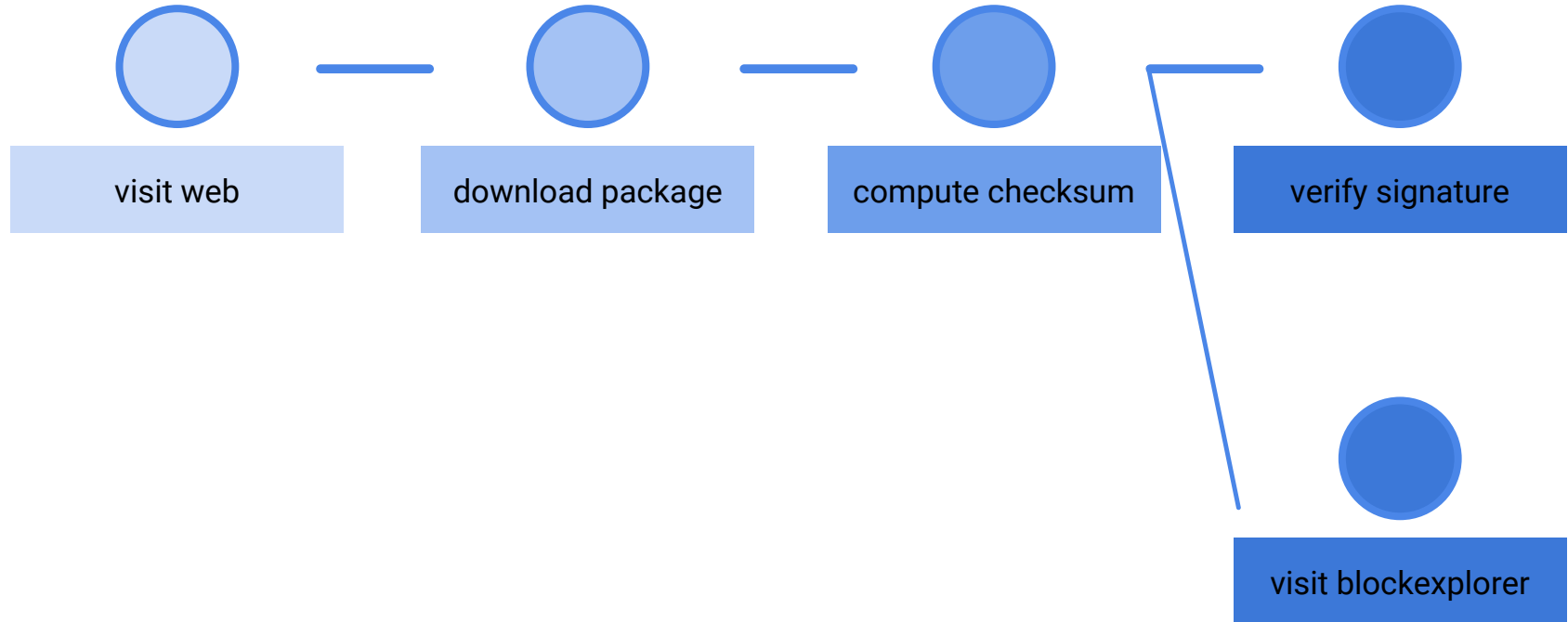
M.S. Astrophysics, Ph.D. Candidate



Developer



End User



ECC PubKey to BTC 1addr

Network ID Byte:
Main Network: 0x00
Test Network: 0x6f
Namecoin Net: 0x34

1 20 bytes

sha256(sha256(1 20 bytes))

32 bytes

Checksum

25-byte binary address

1 20 bytes 4

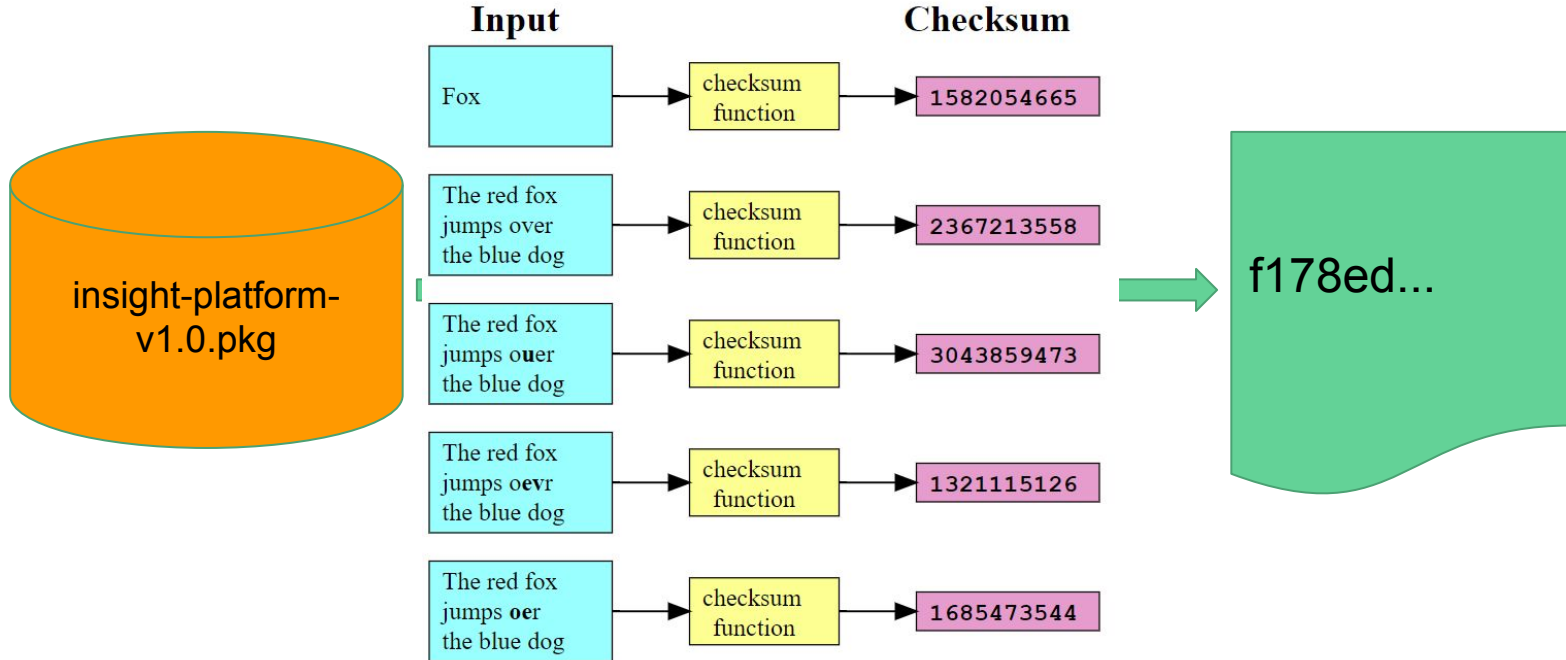
Base256-to-Base58 conversion*
(treat both quantities like big-endian)

1AGRxqDa5WjUKBwHB9XEjmkv1ucoUUy1s

Let's write our data
here

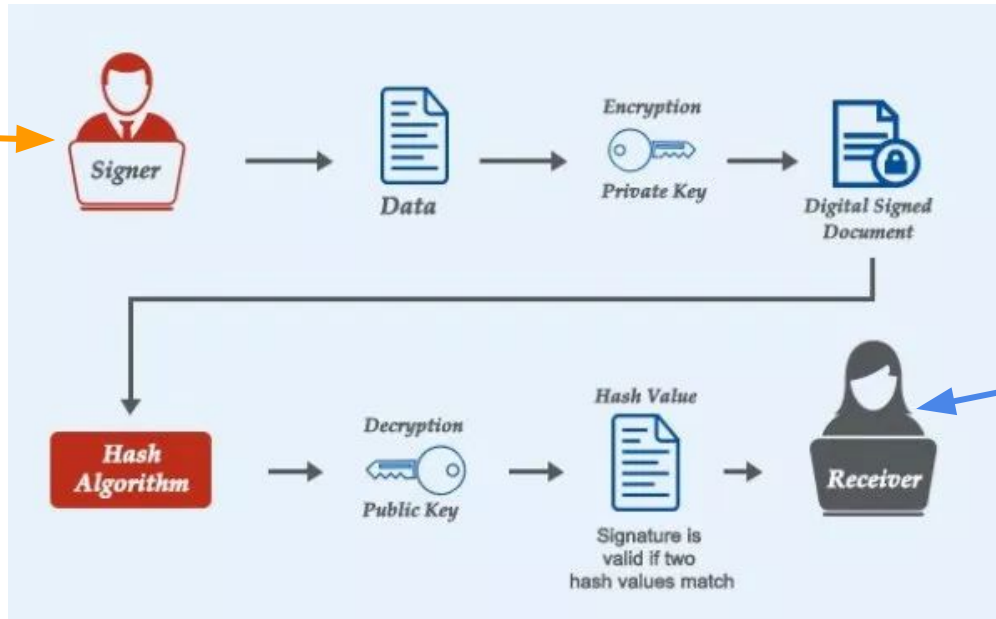
*in a standard base conversion, the 0x00 byte on the left would be irrelevant (like writing '052' instead of just '52'), but in the BTC network the left most zero chars are carried through the conversion. So for every 0x00 byte on the left end of the binary address, we will attach one '1' character to the Base58 address. This is why main-network addresses all start with '1'

Developer and User



Asymmetric Cryptography and Digital Signatures

Developer



End User

OUTSIDE SCOPE OF DEMO

Step 1: Send 1000 Satoshis to a vanity mined developer address:

1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG

Step 2: Checking this address should now show a few UTXOs

Step 3: Run shasum on your file to output a checksum
e.g. run 'shasum 256 insight-platform-v1.0.pkg'

shasum: 256:
f178ed7f17154b292f7bc06aba2e35dcfb467d39

Step 4: Compute a Bitcoin address that encodes this information:
1P1niSKDvNw7VCAA5FSyws95fDF9XMBgHH

Step 5: Generate a raw tx, consuming an UTXO from Step #1 to send
550 Satoshis from:

| | |
|-------------------|------------------------------------|
| Vanity Address: | 1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG |
| to | |
| Checksum Address: | 1P1niSKDvNw7VCAA5FSyws95fDF9XMBgHH |

0200000001f154cba4b162f349e95f29711a58a1596ce1fe5c40577a204cb47c286b8c2bdd010000008
a47304402204ec8909662f91b1191bfe75307afe88bc9f0a171136456caaddbb454647e8dd202205064
fe191a52c9acf721ceff9d02363a6659ca08716c111e20ce1007f9e5d83201410441a044fa4d9ece38d
4076695e645a94478cfad3b6007661db780c6f568eea3d4795908005fd5b3b9d084fe02825ff13964a5
51f7c0af6c54800934f3aa81e31cfdffffff022602000000000000001976a914f178ed7f17154b292f7bc
06aba2e35dcfb467d3988ace421000000000000001976a91485cb5deeca0785dbe22ca4b7b02dc0396d96
df6988ac11b70800

Step 7: Broadcast the signed tx to the network and wait for it
to be mined(confirmation).

Include a link to the transaction on any block explorer.
5b42ea02de5a7d42336d97794c0c86c79540a5a768eccd5d7ee7b9a217bece88

Step 1: Send a 550sat + enough to cover fee to developer address
Preferrably multiple UTXOs (split a large amount into lots of
fixed size UTXOs).

1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG

Step 2: Checking this address should now show a few UTXOs

Step 3: Run shasum on your file

shasum: 256:

Step 4: Compute the address that will encode this shasum:

f178ed7f17154b292f7bc06aba2e35dcfb467d39

Step 5: Use/spend a UTXO to send 550satoshis.

1DCSVhKLR6jyaGa7PGaBmXj5NFtZGe39uG

Step 6: Generate raw tx from ^ to v

1P1niSKDvNw7VCAA5FSywS95fDF9XMBgHH <-- address from step 4

Step 7: Broadcast the tx to the network

Include a link to the transaction on any block explorer.

Justins-MacBook-Pro:demo jmo\$ █

Checksum Example



Via Direct Download

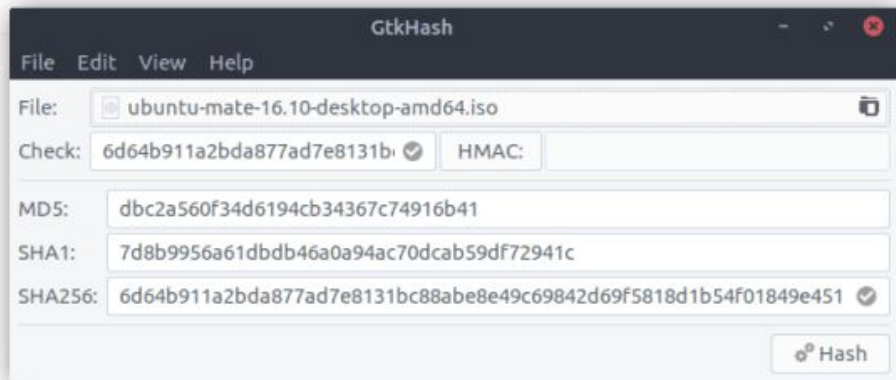
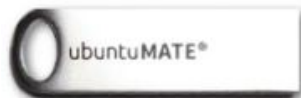
If preferred, you can also download the images over HTTP.

 `ubuntu-mate-16.10-desktop-amd64.iso`

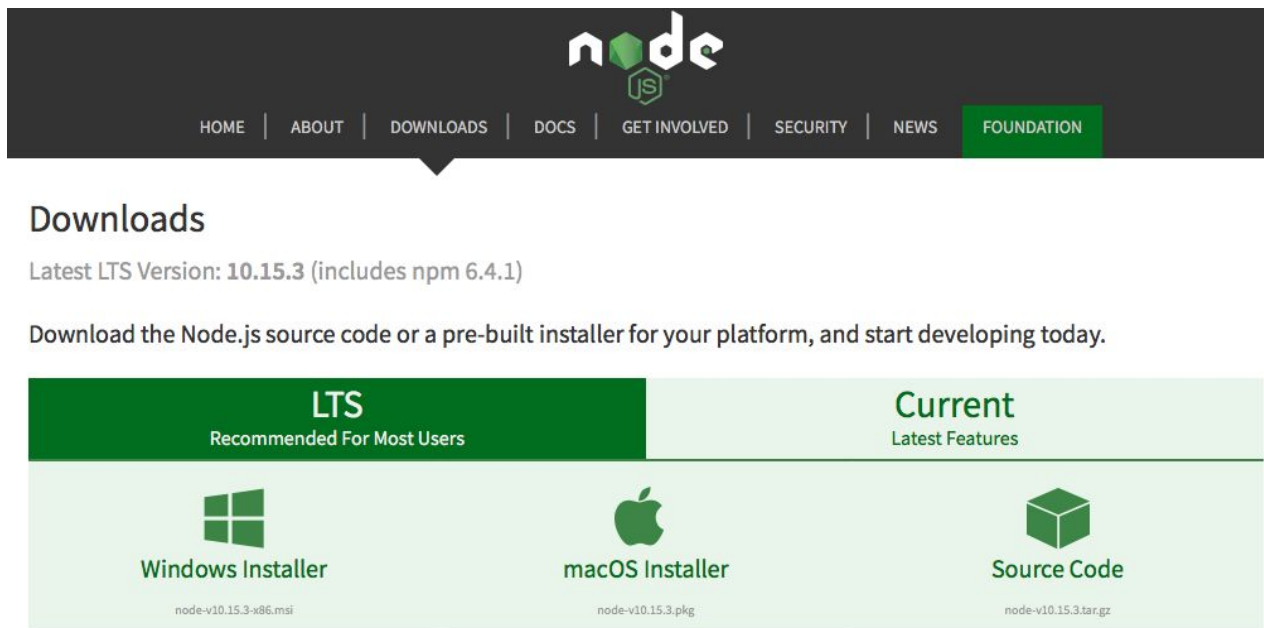
Download Size: 1.7 GB

SHA256 Checksum: `6d64b911a2bda877ad7e8131bc88abe8e49c69842d69f5818d1b54f01849e451`

 [How to verify downloads](#)



NodeJS (Checksum + Signature)



The screenshot shows the Node.js website's 'Downloads' section. At the top is a dark navigation bar with the Node.js logo and links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, NEWS, and FOUNDATION. Below this, the 'Downloads' heading is followed by the text 'Latest LTS Version: 10.15.3 (includes npm 6.4.1)'. A paragraph encourages downloading the source code or a pre-built installer. Two tabs, 'LTS' (Recommended For Most Users) and 'Current' (Latest Features), are visible. Under the 'LTS' tab, three options are listed: 'Windows Installer' (node-v10.15.3-x86.msi), 'macOS Installer' (node-v10.15.3.pkg), and 'Source Code' (node-v10.15.3.tar.gz).




node

HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | NEWS | FOUNDATION

Downloads

Latest LTS Version: **10.15.3** (includes npm 6.4.1)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.

| LTS Recommended For Most Users | Current Latest Features | |
|--|--|---|
|  Windows Installer node-v10.15.3-x86.msi |  macOS Installer node-v10.15.3.pkg |  Source Code node-v10.15.3.tar.gz |

NodeJS

Additional Platforms

SmartOS Binaries

64-bit

Docker Image

Official Node.js Docker Image

Linux on Power Systems

64-bit

Linux on System z

64-bit

AIX on Power Systems

64-bit

- Signed SHASUMS for release files (How to verify)

NodeJS

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA256

f2f018418b6bfa263ec981f04f3fa5337724edae8d77fc3951cd36667ee720ea
7a5eaa1f69614375a695cc62017248e5dccc15b0b8edffa7db5b52997cf992ba
8e3df823a58c7b2de327540a0b57a9bcbf3f706108fe65c4cde9a073caae68cee
9e97ee69072836bfbf2a85c4af627ed152574c30c5a32e40fbfcdfda8d9b562e
f690b880cccf8eb5959436073717b648e4bdc521e3217ab7092d5c033326f6133
c82cd99e01f6e26830f0b3e0465f12f92957ebd69a68c91c03228c2669104359
3d7abbbf64bfbf07c55168ca0f1c17be12b0d93affe9b6cadd39724649215fab9
72529b6f77d95f9422f6d1c6b88c1f921b00e5500alc3ea05927f1ae3704133d
94432c2944fc78c2d5e82103f73596a060451330839562c04c414067007c5997
6958551264884cd479f15ed8d40673655a283ed3bd8552d04e8531cd3ccdf483
af2106b08f68e0884caa505ea7e695facc5b4cd356f1e08258899e94cc4c5df0
0544b08467384ba3b3a462d8907d12cea71ac371f3d118057905dd845be43aad
a2fcc2e1827d7a034f39aad8225b4dd72376ad19f7a7884645a512aeedf4ab5
0736e2ad4e3a7580d87e5b70b9c1ce785b15e849dfdf4f2f846c3039ad1e116c
545caa31bf06b150861ca3a2b1f5112aa92bb855de20fd98f8b7bc3f4c4311d7
6c35b85a7cd4188ab7578354277b2b2ca43eacc864a2a16b3669753ec2369d52
faddbe418064baf2226c2fcbcd038c3ef4ae6f936eb952a1138c7ff8cfe862438
f4d0b944618afae2835b500e0cc1c5a013912597fce5560cd4bcb534f5270754
c678b8e5a2d652f920c1093e6249b08e4746c2d37a5b9f719d04f3243776fb01
3732ae66ad564c192ff3a4a6e66e0d8922823c128bb8a6766ece87226982ad54
db460a63d057ac015b75bb6a879fcbce2fefaaf22afa4b6f6445b9db61ce2270d
4e22d926f054150002055474e452ed6cbb85860aa7dc5422213a2002ed9791d5
9df98cac063229aca443c040fd342a96667891bb8eda821d10aa4d49347d7add
93c881fcd0455a932dd5b06a7a03df27d9fe36155c1d3f351ebfa4e20bflc0d
597a372964252daaba4cb8dcac57305f79cffeeca579625f0cd6ab85d29ccdda
fc28bbdb08b3d9b621c7c0ecd2b42506ca2f356f31f2b64210f413b34cfff31799
46b3d03c96de0b9e7d3a204c6f772759283221f5e58ac225df813076a65e2738
e73398cde3e054da7a0a05a86aa512a47a24b961b0659be30a0f01606ca234a9
a921d1a4fa463e877087b3f25abd0ab05b63489bfffcc9ff47acbbbee4elb7494
4ed045aelba046506948b8f90c02716178cb0084f3b56866ac8d23b591e83235
538c8cc4e0b93facb9d63ed6c55d765ec33a18dd264c6c8b9415ad242521d8e6
525ea4adfd5c166076b273db6c0803283c57c4116fce56229ce87c8eb9fcd25
39efb2a884d2f73680b986534eed000017ce16993ea9d695351593fffb9a7bb34
efed715422fcb7032290ec3c7e3b324126e082ee3a87d6ac497f6c97549e478e
38775185b6f6c090e7039ea0b3e630f4ab83e5c259d8d94f0f35f04ec12c0e98
1848e05e130dda3c3b53830cb78c4b28c137c7aac0890b70a8c63798c332ed5

-----BEGIN PGP SIGNATURE-----

iQEzBAEBCAAAFiEETtd49TnjY0x3nIfG1wYOSKGrAFwFAlx+rpCAGqQ1wYOSKGr
AFzn6Qf/eLpfPexv+5ahnPg8ChYib9GKgNejkewhx9f/DOZ6vSj2Nod35bjoKrCk
/p1FRWIdfs28ZqhVd4LQeWBxYVgdyRTM7zA6oLuNzwCxjuK0vwnqZ2oq+LLnEa15
bsLAWdJJs3mEwvhNh2nQUjt1XFPQDR+cBosuEARSoUlgSqUzfw053x4eyJD5NcE77
944xFi7uob828J2wMebM1L5w0MHIRgqs9ptDHEigERGw4JfakKXhpsT9gT9Igfzx
10FQH50d5z7dZEW019GfOVzsSr0kGf89VVX6y+b/nS8q4mG6x5LZKL+CecEofCr7
pVileZXVzWQnVbdsTPXHNf/o0AHuCW==
=X9UW

-----END PGP SIGNATURE-----

node-v10.15.3-aix-ppc64.tar.gz
node-v10.15.3-darwin-x64.tar.gz
node-v10.15.3-darwin-x64.tar.xz
node-v10.15.3-headers.tar.gz
node-v10.15.3-headers.tar.xz
node-v10.15.3-linux-arm64.tar.gz
node-v10.15.3-linux-arm64.tar.xz
node-v10.15.3-linux-armv6l.tar.gz
node-v10.15.3-linux-armv6l.tar.xz
node-v10.15.3-linux-armv7l.tar.gz
node-v10.15.3-linux-armv7l.tar.xz
node-v10.15.3-linux-ppc64le.tar.gz
node-v10.15.3-linux-ppc64le.tar.xz
node-v10.15.3-linux-s390x.tar.gz
node-v10.15.3-linux-s390x.tar.xz
node-v10.15.3-linux-x64.tar.gz
node-v10.15.3-linux-x64.tar.xz
node-v10.15.3.pkg
node-v10.15.3-sunos-x64.tar.gz
node-v10.15.3-sunos-x64.tar.xz
node-v10.15.3.tar.gz
node-v10.15.3.tar.xz
node-v10.15.3-win-x64.7z
node-v10.15.3-win-x64.zip
node-v10.15.3-win-x86.7z
node-v10.15.3-win-x86.zip
node-v10.15.3-x64.msi
node-v10.15.3-x86.msi
win-x64/node.exe
win-x64/node.lib
win-x64/node_pdb.7z
win-x64/node_pdb.zip
win-x86/node.exe
win-x86/node.lib
win-x86/node_pdb.7z
win-x86/node_pdb.zip

NodeJS

Source Code

node-v10.15.3.tar.gz

Additional Platforms

SmartOS Binaries

64-bit

Docker Image

Official Node.js Docker Image

Linux on Power Systems

64-bit

Linux on System z

64-bit

AIX on Power Systems

64-bit

- Signed SHASUMS for release files (How to verify)

NodeJS

Verifying Binaries

Download directories contain a `SHASUMS256.txt` file with SHA checksums for the files.

To download `SHASUMS256.txt` using `curl`:

```
$ curl -O https://nodejs.org/dist/vx.y.z/SHASUMS256.txt
```

To check that a downloaded file matches the checksum, run it through `sha256sum` with a command such as:

```
$ grep node-vx.y.z.tar.gz SHASUMS256.txt | sha256sum -c -
```

For Current and LTS, the GPG detached signature of `SHASUMS256.txt` is in `SHASUMS256.txt.sig`. You can use it with `gpg` to verify the integrity of `SHASUMS256.txt`. You will first need to import [the GPG keys of individuals authorized to create releases](#). To import the keys:

```
$ gpg --keyserver pool.sks-keyservers.net --recv-keys DD8F2338BAE7501E3DD5AC78C273792F7D83545D
```

See the bottom of this README for a full script to import active release keys.

Next, download the `SHASUMS256.txt.sig` for the release:

```
$ curl -O https://nodejs.org/dist/vx.y.z/SHASUMS256.txt.sig
```

Then use `gpg --verify SHASUMS256.txt.sig SHASUMS256.txt` to verify the file's signature.