Security improvements in TeX Live

Norbert Preining
TeX Live Team
TUG 2016, Toronto
Overview

- status up to (and including) 2015
- possible attack vectors
- integrity and authenticity
- verification architecture
- (non-)distributing GnuGP (and alternatives)

Problems

- user experience
- key management
Status up to 2015

- container checksum (md5) is available in the tlpdb
  
  name 12many
  ...
  containersize 2100
  containermd5 ...
  doccontainersize 375404
  doccontainermd5 ...
  ...

- but ...
Status up to 2015

- container checksum (md5) is available in the tlpdb

  name 12many
  ...
  containersize 2100
  containermd5 ..... 
  doccontainersize 375404
  doccontainermd5 .... 
  ...

- but ... only used to restart an interrupted installation
Status up to 2015

- container checksum (md5) is available in the tlpdb
  name 12many
  ...
  containersize 2100
  containermd5 .....  
  doccontainersize 375404
  doccontainermd5 ....
  ...

- but ... only used to restart an interrupted installation not for \texttt{tlmgr update} nor for normal installation!
Do we need better security?
Possible attack vector I

- compromise one CTAN mirror
Possible attack vector I

- compromise one CTAN mirror
- exchange pdftex binary with one shipping a crypto-virus
Possible attack vector I

- compromise one CTAN mirror
- exchange `pdflatex` binary with one shipping a crypto-virus
- enjoy ...
Possible attack vector I

- compromise one CTAN mirror
- exchange \texttt{pdflatex} binary with one shipping a crypto-virus
- enjoy ...

Since no checks are done, this is easily possible!
Possible attack vector I

- compromise one CTAN mirror
- exchange pdftex binary with one shipping a crypto-virus
- enjoy ...

Since no checks are done, this is easily possible!

Verification of checksums (md5)
In tlcritical for a few months before TL2016 was released, but not pushed out to 2015.
Possible attack vectors II

- compromise one CTAN mirror
Possible attack vectors II

- compromise one CTAN mirror
- exchange `pdftex` binary with one shipping a crypto-virus
Possible attack vectors II

- compromise one CTAN mirror
- exchange `pdftex` binary with one shipping a crypto-virus
- adjust the container so that the MD5 sum does not change (possible!)
Possible attack vectors II

- compromise one CTAN mirror
- exchange `pdftex` binary with one shipping a crypto-virus
- adjust the container that the MD5 sum does not change (possible!)
- enjoy …
Possible attack vectors II

- compromise one CTAN mirror
- exchange \texttt{pdftex} binary with one shipping a crypto-virus
- adjust the container that the MD5 sum does not change (possible!)
- enjoy ...

No counter measures up to \TeX\ Live 2015!
Possible attack vector III

- compromise one CTAN mirror (or setup one yourself, get good connections and many users)
Possible attack vector III

▶ compromise one CTAN mirror (or setup one yourself, get good connections and many users)
▶ exchange pdftex binary as before
Possible attack vector III

▷ compromise one CTAN mirror (or setup one yourself, get good connections and many users)
▷ exchange pdftex binary as before
▷ adjust the checksum in the \texttt{tlpdb} file
Possible attack vector III

- compromise one CTAN mirror (or setup one yourself, get good connections and many users)
- exchange pdftex binary as before
- adjust the checksum in the t1pdb file
- enjoy …
Possible attack vector III

- compromise one CTAN mirror (or setup one yourself, get good connections and many users)
- exchange `pdftex` binary as before
- adjust the checksum in the `tlpdb` file
- enjoy ...

No counter measures up to TeX Live 2015!
Integrity and authenticity

- Integrity: Need to check the integrity of the downloaded packages – prevent tampering.
  - md5 is not strong, can be tampered – switch to sha512
- Authenticity: Verify that the packages are actually the ones from us (TEX Live Team).
  - Cryptographic signatures
Integrity and authenticity

Integrity

Need to check the integrity of the downloaded packages – prevent tampering.
Integrity and authenticity

**Integrity**

Need to check the integrity of the downloaded packages – prevent tampering.

MD5 is not strong, can be tampered
Integrity and authenticity

Integrity

Need to check the integrity of the downloaded packages – prevent tampering.

MD5 is not strong, can be tampered – switch to SHA512
Integrity and authenticity

Integrity

Need to check the integrity of the downloaded packages – prevent tampering.

MD5 is not strong, can be tampered – switch to SHA512

Authenticity

Verify that the packages are actually the ones from us (TEX Live Team).
Integrity and authenticity

**Integrity**

Need to check the integrity of the downloaded packages - prevent tampering.

MD5 is not strong, can be tampered - switch to SHA512

**Authenticity**

Verify that the packages are actually the ones from us (TeX Live Team).

Cryptographic signatures
Verification architecture – overview
Verification architecture – overview

\texttt{tlmgr downloads remote texlive.tlpdb}
Verification architecture – overview

\texttt{tlmgr} downloads remote \texttt{texlive.tlpdb}

\texttt{tlmgr verifies authenticity of the tlpdb}
Verification architecture – overview

\texttt{tlmgr} downloads remote \texttt{texlive.tlpdb}

\texttt{tlmgr} \textbf{verifies authenticity} of the \texttt{tlpdb}

\texttt{tlmgr} \textbf{checks integrity} of containers
Verification architecture – overview

`tlmgr downloads remote texlive.tlpdb`

`tlmgr verifies authenticity of the tlpdb`

`tlmgr checks integrity of containers`

`tlmgr installs package`
Verification architecture – overview

```
\texttt{tlmgr downloads remote texlive.tlpdb}
```

```
\texttt{tlmgr verifies authenticity of the \texttt{tlpdb}}
```

```
\texttt{tlmgr checks integrity of containers}
```

```
\texttt{tlmgr installs package}
```

new

partially new
Verification of authenticity
Verification of authenticity

texlive.tlpdb

name 00texlive.config
...

name 12many
containerchecksum ...
...

name 2up
containerchecksum ...
...
Verification of authenticity

texlive.tlpdb

name 00texlive.config
...
name 12many
containerchecksum ...
...
name 2up
containerchecksum ...
...

texlive.tlpdb.sha512

<128 hex digits> texlive.tlpdb
Verification of authenticity

texlive.tlpdb

name 00texlive.config ...

name 12many containerchecksum ...

name 2up containerchecksum ...

.texlive.tlpdb

<128 hex digits> texlive.tlpdb

texlive.tlpdb.sha512.asc

—–BEGIN PGP SIGNATURE—–
iQEVAwUBVyAV9kzhh3...
r2mB9xEnR4o2SRBDNI...

—–END PGP SIGNATURE—–
Signing key

<table>
<thead>
<tr>
<th>pub</th>
<th>2048R/06BAB6BC 2016-03-19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key fingerprint</td>
<td>C78B 82D8 C795 12F7 9CC0  D7C8 0D5E 5D91 06BA B6BC</td>
</tr>
<tr>
<td>uid</td>
<td>TeX Live Distribution <a href="mailto:tex-live@tug.org">tex-live@tug.org</a></td>
</tr>
<tr>
<td>sig 3</td>
<td>06BAB6BC 2016-03-19 TeX Live Distribution <a href="mailto:tex-live@tug.org">tex-live@tug.org</a></td>
</tr>
<tr>
<td>sig 3</td>
<td>06BAB6BC 2016-03-19 TeX Live Distribution <a href="mailto:tex-live@tug.org">tex-live@tug.org</a></td>
</tr>
<tr>
<td>sig</td>
<td>860CDC13 2016-03-20 Norbert Preining <a href="mailto:norbert@preining.info">norbert@preining.info</a></td>
</tr>
<tr>
<td>sig</td>
<td>30D155AD 2016-03-20 Karl Berry <a href="mailto:karl@freefriends.org">karl@freefriends.org</a></td>
</tr>
</tbody>
</table>

- signed by Karl and my key (mine is also in the Debian keyring)
- actual signing subkey is used, main key is offline
  (in case of breach of TUG server we can revoke the sub-key)
Verification of authenticity II

Why not sign directly?

- Speed up of verification (factor 10)
  - (because this is how I copied it from Debian)
  - Might not be needed (0.01s versus 0.1s)?

- Why sha512?
  - Currently considered uncompromisable (in contrast to md5)
  - Will hopefully hold for several years (other options sha256 etc)
Verification of authenticity II

Why not sign directly?

- speed up of verification (factor 10)
Verification of authenticity II

Why not sign directly?

- speed up of verification (factor 10)
- (because this is how I copied it from Debian)
Verification of authenticity II
Why not sign directly?

- speed up of verification (factor 10)
- (because this is how I copied it from Debian) might not be needed (0.01s versus 0.1s)?
Verification of authenticity II

Why not sign directly?

- speed up of verification (factor 10)
- (because this is how I copied it from Debian) might not be needed (0.01s versus 0.1s)?

Why SHA512?

TEX Live 2016
Verification of authenticity II

Why not sign directly?

- speed up of verification (factor 10)
- (because this is how I copied it from Debian) might not be needed (0.01s versus 0.1s)?

Why SHA512?

- currently considered uncompromisable (in contrast to MD5)
- will hopefully hold for several years (other options SHA256 etc)
Check of integrity

Check the SHA512 checksum of the containers against the (verified) information in the texlive.tlpdb.
Check of integrity

Check the SHA512 checksum of the containers against the (verified) information in the texlive.tlpdb.

Comments

- Why sufficient?
Check of integrity

Check the SHA512 checksum of the containers against the (verified) information in the `texlive.tlpdb`.

Comments

- Why sufficient? — `texlive.tlpdb` gives authenticated information
Check of integrity

Check the SHA512 checksum of the containers against the (verified) information in the texlive.tlpdb.

Comments

- Why sufficient? — texlive.tlpdb gives authenticated information
- We actually check also the size (might delete that one!)
(Non-)distributing of GnuPG

Why not include GnuPG into TeX Live?
(Non-)distributing of GnuPG

Why not include GnuPG into TeX Live?

- We don’t want to support (and compile it)
(Non-)distributing of GnuPG

Why not include GnuPG into \TeX\ Live?

- We don’t want to support (and compile it)
  (but could go into private space like \texttt{xz} and \texttt{wget}!)
(Non-)distributing of GnuPG

Why not include GnuPG into \TeX\ Live?

- We don’t want to support (and compile it) (but could go into private space like \texttt{xz} and \texttt{wget}!)
- Export and import restrictions, Waasenaar Agreement
  Export might be ok nowadays, but there are many countries the strictly forbid \textit{import} of cryptographic software (India, France is a bit unclear, …)
  \textsc{TUG} does not want to get involved in legal battles (not funny) when sending DVDs to India or other countries.
Alternative for \TeX\ Live

\texttt{tlmgr \textit{\textendash}repository http://www.preining.info/tlgpg/}
\texttt{install tlgpg}
Alternative for TeX Live

tlmgr -repository http://www.preining.info/tlgpg/
install tlgpg

▶ installs binaries into tlpkg/installer/gpg/
▶ GnuPG binaries for Windows and Mac (both archs)
▶ already supported by TLU on Mac
▶ most big distributions have GnuPG (1 or 2) installed (both fine)
▶ the TeX Live infrastructure already checks for the above location
▶ not affiliated with TUG (smile)
▶ maybe could be hosted at DANTE or some other server?
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module, but this is not available on older MacOS (shipping 10 years old Perl!)
- Perl/Lua implementation is far too slow (minutes!)
- Solution: try Digest::SHA, openssl, sha512sum, and shasum, one is hopefully available

Users' complains reduce visibility of warnings/information shown, try to provide a unspectacular introduction of the feature
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module, but this is not available on older MacOS (shipping 10 years old Perl!)
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module, but this is not available on older MacOS (shipping 10 years old Perl!)
- Perl/Lua implementation is far too slow (minutes!)
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module, but this is not available on older MacOS (shipping 10 years old Perl!)
- Perl/Lua implementation is far too slow (minutes!)
- Solution: try Digest::SHA, openssl, sha512sum, and shasum, one is hopefully available
Problems

Computing SHA512 checksums

- we use Digest::SHA perl module, but this is not available on older MacOS (shipping 10 years old Perl!)
- Perl/Lua implementation is far too slow (minutes!)
- Solution: try Digest::SHA, openssl, sha512sum, and shasum, one is hopefully available

Users’ complains

reduce visibility of warnings/information shown, try to provide a unspectacular introduction of the feature
User experience – changes in the interface

Aim: nearly no user visible change
User experience – changes in the interface

Aim: nearly no user visible change

[~] tlmgr update --list --repository http://localhost/tlnet/
tlmgr: package repository http://localhost/tlnet/ (verified)
...

...
User experience – changes in the interface

Aim: nearly no user visible change

```bash
[~] tlmgr update --list --repository http://localhost/tlnet/
tlmgr: package repository http://localhost/tlnet/ (verified)
...
```

If not GnuPG is found the output is:

```bash
[~] tlmgr update --list --repository http://localhost/tlnet/
tlmgr: package repository http://localhost/tlnet/ (not verified)
...
```

Similar for multiple repositories
Key management

Aim: support alternative repositories

- `tlmgr key` action added
- allows listing, adding, removing of keys
- already in use (koma script, TeX JP, ...)
Further plans

not many ...
Further plans

not many …

▶ download checksum from TUG for extra protection?
▶ directly sign – reduce one download?
▶ do not check file size (useless overhead)?
Further plans

not many ...

- download checksum from TUG for extra protection?
- directly sign – reduce one download?
- do not check file size (useless overhead)?
- ... your suggestions?
Further plans

not many ...

- download checksum from TUG for extra protection?
- directly sign - reduce one download?
- do not check file size (useless overhead)?
- ... your suggestions?

Thanks for your attention