Subject: TinyVZ 0.7 released
Posted by samiam on Sat, 20 Aug 2011 22:36:28 GMT
View Forum Message <> Reply to Message

I have released TinyVZ 0.7 today. This will be my last TinyVZ release for the foreseeable future.

TinyVZ 0.7 is a tiny little OpenVZ template for making OpenVZ containers that use the lowest amount of memory and hard disk space possible.

This is a self-hosting template with all source code; it is possible to compile the entire system inside of the template. Look in the build/ directory (inside the template) for source code.

The main addition to this release of TinyVZ is that it is now possible to use the relevant vzctl commands to add or remove an IP, set the machine's hostname, determine what nameservers to use, as well as setting user's passwords. As recently discussed on the list, I had to add the Bash shell to do this.

The tarball now contains the template's tarball inside of it; the relevant scripts used by vzctl as well as an installation guide (README) are also inside the tarball.

The system is for hard core UNIX/Linux gurus: The only editor is a miniature version of vi included with Busybox (actually, I also compiled in Busybox's version of the "ed" editor, for those who feel vi pampers the user too much); all configuration is done by editing text files. You will need to compile your own mail server, SSH server, web server, or other desired server.

It can be downloaded here:

http://samiam.org/TinyVZ/

- Sam

Subject: Re: TinyVZ 0.7 released
Posted by Gary Wallis on Sat, 20 Aug 2011 22:54:06 GMT
View Forum Message <> Reply to Message

Thanks, GREAT WORK!

Subject: Re: TinyVZ 0.7 released

On Sun, Aug 21, 2011 at 12:36 AM, Sam Trenholme
<strenholme.usenet@gmail.com> wrote:
> I have released TinyVZ 0.7 today. This will be my last TinyVZ release
> for the foreseeable future.
>
> TinyVZ 0.7 is a tiny little OpenVZ template for making OpenVZ
> containers that use the lowest amount of memory and hard disk space
> possible.
>
> This is a self-hosting template with all source code; it is possible
> to compile the entire system inside of the template. Look in the
> build/ directory (inside the template) for source code.
>
> The main addition to this release of TinyVZ is that it is now possible
> to use the relevant vzctl commands to add or remove an IP, set the
> machine's hostname, determine what nameservers to use, as well as
> setting user's passwords. As recently discussed on the list, I had to
> add the Bash shell to do this.
>
> The tarball now contains the template's tarball inside of it; the
> relevant scripts used by vzctl as well as an installation guide
> (README) are also inside the tarball.
>
> The system is for hard core UNIX/Linux gurus: The only editor is a
> miniature version of vi included with Busybox (actually, I also
> compiled in Busybox's version of the "ed" editor, for those who feel
> vi pampers the user too much); all configuration is done by editing
> text files. You will need to compile your own mail server, SSH
> server, web server, or other desired server.
>
> It can be downloaded here:
>
> http://samiam.org/TinyVZ/

Just had a quick try, xz compression does not seem to be supported by
vzctl, had to recompress in tar.gz format:

root@mybox /root/zoobab/tmp/TinyVZ-0.7 [51]# vzctl create 889
--ostemplate TinyVZ-0.7-template
Creating container private area (TinyVZ-0.7-template)
Cached OS template
/var/lib/vz/template/cache/TinyVZ-0.7-template.tar.gz not found
Creation of container private area failed

--
Benjamin Henrion <bhenrion at ffii.org>

FFII Brussels - +32-484-566109 - +32-2-4148403
"In July 2005, after several failed attempts to legalise software
patents in Europe, the patent establishment changed its strategy.
Instead of explicitly seeking to sanction the patentability of
software, they are now seeking to create a central European patent
court, which would establish and enforce patentability rules in their
favor, without any possibility of correction by competing courts or
democratically elected legislators."

Subject: Re: TinyVZ 0.7 released
Posted by kir on Sun, 21 Aug 2011 11:56:12 GMT
View Forum Message <> Reply to Message

On 08/21/2011 03:44 PM, Benjamin Henrion wrote:
> Just had a quick try, xz compression does not seem to be supported by
> vzctl, had to recompress in tar.gz format:

.xz templates are supported since vzctl-3.0.28, released in June.

Subject: Re: TinyVZ 0.7 released
Posted by Benjamin Henrion on Sun, 21 Aug 2011 13:49:14 GMT
View Forum Message <> Reply to Message

On Sun, Aug 21, 2011 at 12:36 AM, Sam Trenholme
<strenholme.usenet@gmail.com> wrote:
> I have released TinyVZ 0.7 today. This will be my last TinyVZ release
> for the foreseeable future.
>
> TinyVZ 0.7 is a tiny little OpenVZ template for making OpenVZ
> containers that use the lowest amount of memory and hard disk space
> possible.
>
> This is a self-hosting template with all source code; it is possible
> to compile the entire system inside of the template. Look in the
> build/ directory (inside the template) for source code.
>
> The main addition to this release of TinyVZ is that it is now possible
> to use the relevant vzctl commands to add or remove an IP, set the
> machine's hostname, determine what nameservers to use, as well as
> setting user's passwords. As recently discussed on the list, I had to
> add the Bash shell to do this.
>
> The tarball now contains the template's tarball inside of it; the
> relevant scripts used by vzctl as well as an installation guide
> (README) are also inside the tarball.

>
> The system is for hard core UNIX/Linux gurus: The only editor is a
> miniature version of vi included with Busybox (actually, I also
> compiled in Busybox's version of the "ed" editor, for those who feel
> vi pampers the user too much); all configuration is done by editing
> text files. You will need to compile your own mail server, SSH
> server, web server, or other desired server.
>
> It can be downloaded here:
>
> http://samiam.org/TinyVZ/

I just took your bash binary and put into an Openwrt backfire rootfs,
fixed some pts mount, and here it is:

```
=====================================================
root@molotov /var/lib/vz/private [249]# vzctl enter 889
entered into CT 889

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 Backfire (10.03.1-RC5, r27608) -------------------------
  * 1/3 shot Kahlua     In a shot glass, layer Kahlua
  * 1/3 shot Bailey's   on the bottom, then Bailey's,
  * 1/3 shot Vodka      then Vodka.
 ----------------------------------------------------
root@OpenWrt:/#
=====================================================
```

Now I have to create the dists file and scripts to make the network working.

I am on #openvz channel if you want to join.

The memory consumption with openwrt are quite similar, but it is more
extensible.

--
Benjamin Henrion <bhenrion at ffii.org>
FFII Brussels - +32-484-566109 - +32-2-4148403
"In July 2005, after several failed attempts to legalise software
patents in Europe, the patent establishment changed its strategy.
Instead of explicitly seeking to sanction the patentability of
software, they are now seeking to create a central European patent
court, which would establish and enforce patentability rules in their
favor, without any possibility of correction by competing courts or
democratically elected legislators."

Subject: Re: TinyVZ 0.7 released
Posted by kir on Sun, 21 Aug 2011 14:12:16 GMT
View Forum Message <> Reply to Message

On 08/21/2011 02:36 AM, Sam Trenholme wrote:
> The tarball now contains the template's tarball inside of it; the
> relevant scripts used by vzctl as well as an installation guide
> (README) are also inside the tarball.

If you like, please submit the vzctl part, I will include it into the
next release.

Subject: Re: TinyVZ 0.7 released
Posted by samiam on Sun, 21 Aug 2011 20:47:00 GMT
View Forum Message <> Reply to Message

How should I submit the relevant scripts for vzctl?

- Sam

2011/8/21 Kir Kolyshkin <kir@openvz.org>:
> On 08/21/2011 02:36 AM, Sam Trenholme wrote:
>>
>> The tarball now contains the template's tarball inside of it; the
>> relevant scripts used by vzctl as well as an installation guide
>> (README) are also inside the tarball.
>
> If you like, please submit the vzctl part, I will include it into the next
> release.
>
>

Subject: Re: TinyVZ 0.7 released
Posted by Solar Designer on Mon, 22 Aug 2011 16:12:18 GMT
View Forum Message <> Reply to Message

On Sat, Aug 20, 2011 at 06:36:28PM -0400, Sam Trenholme wrote:
> TinyVZ 0.7 is a tiny little OpenVZ template for making OpenVZ
> containers that use the lowest amount of memory and hard disk space
> possible.
>
> This is a self-hosting template with all source code; it is possible
> to compile the entire system inside of the template. Look in the
> build/ directory (inside the template) for source code.

This is very nice. My concern, though, is that things such as uClibc

were not built with security in mind. I am pretty sure that uClibc is problematic when used in conjunction with SUID/SGID programs. Does uClibc ensure that fd 0-2 are open on program startup (opening them to /dev/null / /dev/full if not)? I doubt it. I admit I haven't checked, though.

I think a better libc to use for this purpose would be musl:

http://www.etalabs.net/musl/

It also lacks some of those highly desirable security measures, but I think it will gain those soon.

While uClibc is primarily for embedded systems, musl is primarily for typical/full systems - just without glibc's bloat.

Meanwhile, the sad truth may be that under Linux we need to use (e)glibc (or other clones of it) for SUIDs/SGIDs. <plug>BTW, the full Owl userland, with development/build tools, is just 112 MB under .tar.gz:
http://mirrors.kernel.org/openwall/Owl/current/vztemplate/
It is also able to rebuild itself, although the source code is not part of the .tar.gz (just the development/build tools/libs are). With the source tarballs added, the size increases a lot indeed... to 280 MB if we exclude just the Linux kernel, which is not installed in an OpenVZ container anyway.
</plug>

Alexander

Subject: Re: TinyVZ 0.7 released
Posted by samiam on Mon, 22 Aug 2011 18:35:35 GMT
View Forum Message <> Reply to Message

> This is very nice. My concern, though, is that things such as uClibc
> were not built with security in mind. I am pretty sure that uClibc is
> problematic when used in conjunction with SUID/SGID programs.

You know, this is going to make me start yet another almost completely off-topic rant.

Bruce Schneier summed it up best when he said that security is a process, not a product. In other words, security is a process of discovering how to make programs more secure. In the 1980s, programs were so insecure, it was not unheard of for programs to have intentional backdoors in them (the Sendmail "Wiz" command, for example). Soon break-ins were not a simple matter of finding backdoors (or default accounts, such as what the 1980s antagonist in

Cliff Stoll's "The Cuckoo's Egg" used); people had to use buffer overflows, which a lot of 1990s code had.

C programmers have cleaned up their act; not only do skilled programmers have coding styles which avoid these kinds of mistakes, but there are a lot of other tricks to minimize the impact of this kind of error (the NX bit, etc.)

Not even the most skilled programmer in the world is guaranteed to make a perfectly secure program. For a while, advocates of Dr. Bernstein's software believed his software was perfectly secure and never needed to be updated, but while he is a brilliant programmer who makes very few security mistakes in his code, there is still the occasional security bug: djbdns, for example, has three known security bugs. [1]

Bernstein's software has a powerful lesson: For a program to remain secure, it has to be maintained. There has to be someone who is accountable for the security holes in the software and is willing to fix it. This is why I use a derivative of RedHat enterprise Linux: RedHat stands behind their software for seven years and I know I can keep a system reasonably secure for that time duration without having to be on the constant update-by-reinstall treadmill most other Linux distributions keep me on.

One of the reasons I have kept the number of packages in my TinyVZ distribution down to an absolute minimum is because this minimizes the number of potential security holes I will have to babysit in this distribution.

> Does uClibc ensure that fd 0-2 are open on program startup (opening them to
> /dev/null / /dev/full if not)? I doubt it. I admit I haven't checked,
> though.

It might. It might now. I haven't checked either. What I can tell you is that there does not appear to be any vulnerability reports for uClibc:

http://security-tracker.debian.org/tracker/source-package/uc libc

(note to self: There is a vulnerability report for the gzip stuff in the Busybox included with TinyVZ)

> I think a better libc to use for this purpose would be musl:
>
> http://www.etalabs.net/musl/
>
> It also lacks some of those highly desirable security measures, but I

> think it will gain those soon.

If I were to do this again in a few months, I may use musl; however musl is an "alpha" product while uClibc is a mature product that is still being updated. I use the code which works today; that's why I'm using OpenVZ and not, say, LXC [2].

TinyVZ is, in truth, me putting closure on a project I had back in 2007 (around the time I started rewriting MaraDNS's recusrive resolver) making a tiny uClibc + Busybox live CD distribution for having on a business card CD so I could use cyber cafes and friends' infected computers in a reasonably secure manner.

I never distributed that code, mainly because I never fully made it independent from the DeLi distribution it was built on until, by a weekend of hard-core hacking, I made the system self-hosting about a week ago. From there, it was relatively simple to add Bash and write the scripts used by OpenVZ to configure the system with vzctl.

> Meanwhile, the sad truth may be that under Linux we need to use (e)glibc
> (or other clones of it) for SUIDs/SGIDs.

This can very well be true. One thing I have done is minimize the attack surface with SUIDs by having only two SUID programs in the system: "passwd" and "su". While Busybox is supposed to have a way of having it be SUID and drop privileges as needed, I don't fully trust that mechanism; better to compile Busybox twice.

><plug>BTW, the full Owl
> userland, with development/build tools, is just 112 MB under .tar.gz:
> http://mirrors.kernel.org/openwall/Owl/current/vztemplate/
> It is also able to rebuild itself, although the source code is not part
> of the .tar.gz (just the development/build tools/libs are). With the
> source tarballs added, the size increases a lot indeed... to 280 MB if
> we exclude just the Linux kernel, which is not installed in an OpenVZ
> container anyway.
> </plug>

With the full self-hosting development tree being just over 40 megs xz compressed, and a usable "DNS toaster" system (which is resolving all of my DNS queries as I type this) being about 2 megs in size, I think TinyVZ targets those who want to have a really small OpenVZ system. OpenVZ's strength is that it allows a single computer to safely run a dozen or more separate services with full compartmentalization. By making the containers as small as possible, I can visualize a single server rack with a hub inside of it, as well as a dozen or so credit-card sized computers with Atom SOC cpus, 4 gigs of memory, and 64GB SSDs. Each one of those computers can run dozens of really tiny

OpenVZ single-task containers.

Owl looks like a really good distribution, and I think it is very wise to stay in step with RedHat, since then RedHat's security updates can be applied. Does OWL have its own mechanism for applying security patches, or does it just use the patches from CentOS or Scientific Linux [3]?

> Alexander

- Sam

[1] I have blogged about this, see http://aac2.vk.tj

[2] Idiots who say OpenVZ is "deprecated" and insist LXC is the future of Linux containers really annoy me. LXC has not had the extensive security audit OpenVZ has (yes, a lot of LXC's code was contributed by the OpenVZ developers). Changing a solution that works today and that the development team stands behind with constant security updates with one that is new and unproven based upon baseless accusations of it being "deprecated" is not, IMHO, a good idea. Examples: http://ahva.vk.tj http://ahvb.vk.tj

[3] I have blogged about how it is annoying how security updates sometimes aren't made available to CentOS systems: http://acs2.vk.tj. I have also blogged about how to apply Scientific Linux security updates to running CentOS 5 systems, since there isn't a "Scientific Linux 5" OpenVZ template: http://ahi2.vk.tj.

Subject: Re: TinyVZ 0.7 released
Posted by Solar Designer on Mon, 22 Aug 2011 20:44:52 GMT
View Forum Message <> Reply to Message

On Mon, Aug 22, 2011 at 02:35:35PM -0400, Sam Trenholme wrote:
> You know, this is going to make me start yet another almost completely
> off-topic rant.

I like and agree with most of what you wrote.

> One of the reasons I have kept the number of packages in my TinyVZ
> distribution down to an absolute minimum is because this minimizes the
> number of potential security holes I will have to babysit in this
> distribution.

Sure. We do the same thing in Owl - e.g., trying to avoid having more than one implementation of a feature. I think we may start to deviate from this soon, though, due to popular demand... Saying that we don't

have wget in the base system because we have lftp (which has a mirror command, ftp/http/https, and includes lftpget) doesn't always cut it.

> > Does uClibc ensure that fd 0-2 are open on program startup (opening them to
> > /dev/null / /dev/full if not)? I doubt it. I admit I haven't checked,
> > though.
>
> It might. It might now. I haven't checked either. What I can tell
> you is that there does not appear to be any vulnerability reports for
> uClibc:
>
> http://security-tracker.debian.org/tracker/source-package/uc libc

This could mean no security issues, or more likely it could mean that no one cares about "local" security issues, or that upstream does not care about such issues even if some users do. I don't know which it is.

As to musl, I brought the fd 0-2 issue up and as expected Rich is going to fix it:

http://www.openwall.com/lists/musl/2011/08/22/5

> > I think a better libc to use for this purpose would be musl:
> >
> > http://www.etalabs.net/musl/
> >
> > It also lacks some of those highly desirable security measures, but I
> > think it will gain those soon.
>
> If I were to do this again in a few months, I may use musl; however
> musl is an "alpha" product while uClibc is a mature product that is
> still being updated. I use the code which works today; that's why I'm
> using OpenVZ and not, say, LXC [2].

This makes sense.

musl might actually be more mature than LXC, though, if these can be compared at all (apples and oranges, indeed). There's already an experimental desktop distro with X built upon musl.

> TinyVZ is, in truth, me putting closure on a project I had back in
> 2007 (around the time I started rewriting MaraDNS's recusrive
> resolver) making a tiny uClibc + Busybox live CD distribution for
> having on a business card CD so I could use cyber cafes and friends'
> infected computers in a reasonably secure manner.

Hmm, we already had Owl LiveCDs at the time (and much earlier), so you could just use that. ;-) Not for business card sized CDs (for full

CDs), but you could either use a business card sized DVD (which works faster anyway) or exclude /usr/src and a few other optional things.

> > Meanwhile, the sad truth may be that under Linux we need to use (e)glibc
> > (or other clones of it) for SUIDs/SGIDs.
>
> This can very well be true.

I expect that musl will also be a valid option for this very soon.

> One thing I have done is minimize the
> attack surface with SUIDs by having only two SUID programs in the
> system: "passwd" and "su". While Busybox is supposed to have a way of
> having it be SUID and drop privileges as needed, I don't fully trust
> that mechanism; better to compile Busybox twice.

Sounds good. As you might have heard, we have no SUIDs in a default install of Owl (only some SGIDs). And, by the way, musl supports Owl's /etc/tcb shadowing scheme, so you can mix these two and have non-SUID passwd command in a tiny distro without PAM.

> With the full self-hosting development tree being just over 40 megs xz
> compressed, and a usable "DNS toaster" system (which is resolving all
> of my DNS queries as I type this) being about 2 megs in size, I think
> TinyVZ targets those who want to have a really small OpenVZ system.
> OpenVZ's strength is that it allows a single computer to safely run a
> dozen or more separate services with full compartmentalization. By
> making the containers as small as possible, I can visualize a single
> server rack with a hub inside of it, as well as a dozen or so
> credit-card sized computers with Atom SOC cpus, 4 gigs of memory, and
> 64GB SSDs. Each one of those computers can run dozens of really tiny
> OpenVZ single-task containers.

Sure. What you did is very nice.

For now, we're just using OpenVZ containers with instances of Owl, though. That's 112 megs under .tar.gz, not 40, but it is still acceptable for systems with disks.

> Owl looks like a really good distribution, and I think it is very wise
> to stay in step with RedHat, since then RedHat's security updates can
> be applied. Does OWL have its own mechanism for applying security
> patches, or does it just use the patches from CentOS or Scientific
> Linux [3]?

For stuff that is part of Owl (such as everything in our ISOs), we're preparing and making available security and other updates ourselves:

http://openwall.info/wiki/Owl/upgrade

Indeed, we sometimes reuse patches prepared by other distro vendors (and they sometimes reuse ours).

For stuff that a user/admin of Owl might install on top of Owl from another distro's repository, indeed they need to use that distro's updates.

Alexander
