

F5 Friday: What's Inside an F5?



Lori MacVittie, 2012-10-02

Is it Linux? Is it third-party? Is it proprietary? Isn't vcmp just a virtualization platform? Just what is inside an F5 BIG-IP that makes it go vroom?

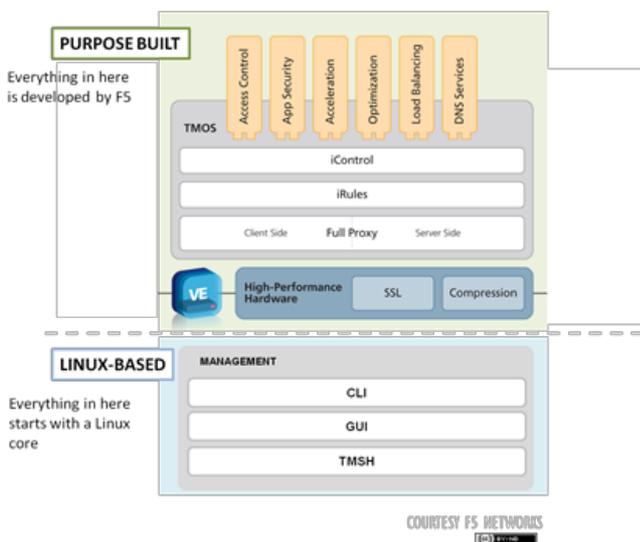


Over the years I've seen some pretty wild claims about what, exactly, is "inside" a BIG-IP that makes it go. I've read articles that claim it's Linux, that it's based on Linux, that it's voodoo magic. I've heard competitors make up information about just about every F5 technology – TMOS, vCMP, iRules – that enables a BIG-IP to do what

it does.

There are two sources of the confusion with respect to what's really inside an F5 BIG-IP. The first stems, I think, from the evolution of the BIG-IP. Once upon a time, BIG-IP was a true appliance – a pure software solution delivered pre-deployed on pretty standard hardware. But it's been many, many years since that was true, since before v9 was introduced back in 2004. BIG-IP version 9 was the beginning of BIG-IP as not a true appliance, but a purpose-built networking device. Appliances deployed on off the shelf hardware generally leverage existing operating systems to manage operating system and even networking tasks – CPU scheduling, I/O, switching, etc... but BIG-IP does not because with version 9 the internal architecture of BIG-IP was redesigned from the ground up to include a variety of not-so-off-the-shelf components. Switch backplanes aren't commonly found in your white-box x86 server, after all, and a bladed chassis isn't something common operating systems handle.

TMOS – the core of the BIG-IP system – is custom built from the ground up. It had to be to support the variety of hardware components included in the system – the FPGAs, the ASICs, the acceleration cards, the switching backplane. It had to be custom built to enable advances in BIG-IP to support the non-disruptive scale of itself when it became available on a chassis-based hardware platform. It had to be custom built so that advances in internal architectures to support virtualization of its compute and network resources, a la vCMP, could come to fruition.



The second source of confusion with respect to the internal architecture of BIG-IP comes from the separation of the operational and traffic management responsibilities. Operational management – administration, configuration, CLI and GUI – resides in its own internal container using off-the-shelf components and software. It's a box in a box, if you will. It doesn't make sense for us – or any vendor, really – to recreate the environment necessary to support a web-based GUI or network access (SSH, etc...) for management purposes. That side of BIG-IP starts with a standard Linux core operating system and is tweaked and modified as necessary to support things like TMSH (TMOS Shell).

That's all it does. Monitoring, management. It generates pretty charts and collects statistics. It's the interface to the configuration of the BIG-IP. It's lights out management. This "side" of BIG-IP has nothing to do with the actual flow of traffic through a BIG-IP aside from configuration. At run time, when traffic flows through a BIG-IP, it's all going through TMOS – through the purpose and very custom built system designed specifically to support application delivery services.

This very purposeful design and development of technology is too often mischaracterized – intentionally or unintentionally – as third-party or just a modified existing kernel/virtualization platform. That's troubling because it hampers the understanding of just what such technologies do and why they're so good at doing it.

Take vCMP, which has sometimes been maligned as little more than third-party virtualization. That's somewhat amusing because vCMP isn't really virtualization in the sense we think about virtualization today. vCMP is designed to allow the resources for a guest instance to span one or multiple blades. It's an extension of multi-processing concepts as applied to virtual machines. If we analogized the technology to server virtualization, vCMP would be the ability to assign compute and network resources from server A to a virtual machine running on server B. Cloud computing providers cannot do this (today) and it's not something that's associated with today's cloud computing models; only grid computing comes close, and it still takes a workload-distributed view rather than a resource-distributed view.

vCMP stands for virtual CMP – clustered multi-processing. CMP was the foundational technology introduced in BIG-IP version 9.4 that allowed TMOS to take advantage of multiple multi-core processors by instantiating one TMM (Traffic Management Microkernel) per core, and then aggregating them – regardless of physical location on BIG-IP – to appear as a single pool of resources. This allowed BIG-IP to scale much more effectively. Basically we applied many of the same high-availability and load distribution techniques we use to ensure applications are fast and available to our internal architecture. This allowed us to scale across blades and is the reason adding (or removing) blades in a VIPRION is non-disruptive.

Along comes a demand for multi-tenancy, resulting in *virtual* CMP. vCMP isn't the virtual machine, it's the technology that manages and provisions BIG-IP hardware resources across multiple instances of BIG-IP virtual machines; the vCMP guests, as we generally call them. What we do under the covers is more akin to an application (a vCMP guest) being comprised of multiple virtual machines (cores), with load balancing providing the mechanism by which resources are assigned (vCMP) than it is simple virtualization.

So now you know a lot more about what's inside a BIG-IP and why we're able to do things with applications and traffic that no one else in the industry can. Because we aren't relying on "standard" virtualization or operating systems. We purposefully design and develop the internal technology specifically for the task at hand, with an eye toward how best to provide a platform on which we can continue to develop technologies that are more efficient and adaptable.

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com