



2018 年 12 月

xService Release 3

安装指南

张文权

POWERSUITE@HOTMAIL.COM

1 概览

1.1 端口服务

端口服务设计目的是以端口来定义服务。传统的网络端口服务，1024 以下被定义为周知端口，如 80 是 http 服务端口，21 是 ftp 服务端口。xService 使用的端口在 1024—65535 之间，这些端口通常称为注册端口，xService 提供的是调度框架服务，包括 xhttp(s)调度服务（基于 http/https 协议），slots 调度服务（基于 tcp/ssl 协议），或者 pnps 调度服务（基于 udp 协议）。注册端口使用何种调度服务由用户指定。

1.1.1 调度服务

与传统的网络服务不同的是 xService 不是专有的、特定的网络服务，比如 IIS 是专业的 WWW 服务。xService 是通用的调度框架，对网络服务来说，可以理解为侦听服务。xService 关心的是调度响应例程的性能，鉴于此，调度服务采用异步 IO 复用模型。在 IO 复用方式上，xService 可以根据操作系统的资源环境来配置 IO 模式和侦听线程的数量，比如在 windows 平台可以使用 IOCompletionPort 模式，linux 平台可以采用 epoll 模式，而 macOS 平台则可以选择 kqueue 模式，除了以上高性能的 IO 复用方式外，也可以使用 select 或 event 模式来复用 IO。为了达到更好的并发性能，侦听的例程可以根据 CPU 的核数来配置合适的侦听线程数量。

1.1.2 服务例程（工作者线程或进程）

服务例程被 xService 调度的一段代码或一个程序，受调例程并不强求是 c/c++ 程序，也可以是其他语言编写的动态库或二进制程序，但他们通常需要暴露一个 API 调用入口作为 xService 的调度入口。对于动态库方式的受调例程，用户可以根据配置来指定受调例程运行于线程或子进程中。运行于子进程中的受调例程理论上在调度速度上会慢于线程调度，但也具有显而易见的优势，一方面子进程相比线程具有天生的健壮性，另一方面即便是 64 位 xService 也可以调度 32 位的子进程。

1.1.3 跨平台性

xService 目前支持 windows、linux、macOS 平台，在 windows 平台，xService 需安装为系统服务，使用 Service Manager 来启停服务。在 linux 和 macOS 平台，xService 则作为 Daemon 值守进程运行。

1.2 调度框架

xService 支持的调度框架为 xhttps (http(s)协议), xslots (tcp/ssl 协议) 和 xpnps (udp 协议)。

1.2.1 xhttps 框架

http(s)是 www 网络服务, 但 xhttps 服务不提供静态网页服务, 可以理解为 xhttps 并非面向浏览器提供服务, 而是面向更广泛的客户桌面程序提供网络应用服务。xhttps 对 xml, soap, json, svg 等通用应用协议均提供高性能的解析支持。同样也支持 RestFul 风格的网络文件服务、关系数据库访问网关和 OSS 存储对象访问代理服务。

访问 xhttps 除了需要 ssl 协议 (TLS3) 的 CA 证书外, 还需要 Authorization 鉴权。因此, 理论上鉴权不通过的 http 访问协议头访问都会被拒绝在 xService 调度服务之外, 这在很大程度上能减轻恶意 tcp 访问冲击端口时耗用系统资源的负荷。

1.2.2 xslots 框架

tcp 是具备可靠传输、流式读写的网络协议, ssl 是基于证书认证的 tcp 安全传输协议。xslots 为基于 tcp/ssl 协议的应用提供调度框架。目前 xService 支持并运行于 xslots 框架下的服务例程有如: mqtt, radhex, pacs dimse, hl7 express。

1.2.3 xpnps 框架

udp 是无可靠传输保证、报文无序、非流式读写 (数据报) 的网络协议, xService 对 udp 协议进行了轻量的封装, 称为 pnp 协议, pnp 保留了 udp 报文高速传输的特点, 并为 udp 报文的传输提供了有序保障, 同时支持流式读写。一次完整的 pnp 会话, 通常需要 1 个启动端口 (注册端口) 和 2 个动态随机端口 (49152—65535)。目前 xService 支持并运行于 xpnps 框架下的服务例程有如: tftp, ring。

1.3 服务例程

xService 支持的服务包括通用服务、数据服务和专业服务。通用服务即常用的网络协议服务, 数据服务通常为关系和非关系数据库提供的访问网关服务。专业服务为一些专有的行业网络应用服务, 如 pacs (医疗影像传输与存储), hl7 express (医疗文档交付与数字胶片存储)。

1.3.1 oau for xhttps

oau for xhttps 是 OAuth 2.0 协议的实现, 它为用户提供了一个单点登录应用群的机制。即为用户提供一个令牌, 而不是用户名和密码来访问他们存放在

特定服务提供者的数据。对于众多第三方服务来说，oau 是他们用户鉴权的代理。

1.3.2 xml for xhttps

xhttps 充分支持 http+xml 的应用，内置高速、流式 xml 解析器，支持超大 xml 文件解析。

1.3.3 soa for xhttps

soa for xhttps 是 SOAP 1.2 协议的实现，xhttps 同样支持 wsdI 文档自动生成。

1.3.4 json for xhttps

http+json 是轻量级远程过程调用的协议，xhttps 内置 json 解析器，并支持 json 和 xml 进行语义一致的转换。

1.3.5 svg for xhttps

svg 是矢量图的描述语言，xhttps 支持将应用表单转换为矢量图以实现面向 h5 应用的可视化输出。

1.3.6 loc for xhttps

loc 服务实现了 http 的共享文件服务，支持分段读写，断点续传。

1.3.7 oss for xhttps

oss 服务实现了远程 oss 对象访问代理。

1.3.8 xdb for xhttps

xdb 是关系数据库访问网关服务，目前支持 ms sqlserver, oracle, mysql 的数据库访问。

1.3.9 mqtt for xslots

mqtt 是物联协议，xslots 支持 mqtt 订阅和推送服务。

1.3.10 radhex for xslots

radhex 是 xService 内置的 KV 内存数据库，rad 服务是 tcp 协议之上的网络服务。

1.3.11 pacs for xslots

pacs 是医疗影像传输和存储服务协议，dimse 是 xslots 框架下的符合 dicom 3.0 的 pacs 网络服务。

1.3.12 hl7 express for xslots

hl7 express 是医疗文档交付和数字胶片存储协议，hl7 express 实现了人与人、人与机器、人与机构、机构与机构之间不同场景下进行医疗文档交付的传输规格、存储规格和可视形式的协议。

1.3.13 tftp for xpnps

实现了 udp 协议之上的文件传输的协议。

1.3.14 ring for xpnps

是一种 udp 协议之上的呼叫信号的传输和定位协议。

2 安装

2.1 安装

2.1.1 windows 平台

文件目录: C:/program files/xService

环境变量: set PATH=%PATH%; C:/program files/xService/api;

服务安装: SC CREATE xService binPath= "C:\Program Files\xService\xService.exe"
start= demand DisplayName= "Easily Port Service"

服务启动: SC START xService

服务停止: SC STOP xService

删除服务: SC DELETE xService

2.1.2 linux 平台

文件目录: /usr/local/xportd

环境变量: /etc/profile 末行添加

export PATH=/usr/local/xportd: /usr/local/xportd/api: \$PATH

服务安装: 启动脚本: /etc/init.d/xportd PID 文件: /var/run/xportd.pid

服务启动: service xportd start

服务停止: service xportd stop

2.1.3 macOS 平台

(暂略)

2.2 配置

2.2.1 配置文件

xService 服务配置文件位于安装文件目录下 xportd.config。

配置样例:

```
<?xml version="1.0" encoding="utf-8"?>
<config>
<port type="http" bind="8889">
<mode>thread</mode>
<root> C:/Program Files/xService</root>
<module>xhttps.exe</module>
```

```
<param>CERT:</param>
</port>
<port type="http" bind="8888">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xhttps.exe</module>
  <param>CERT:SSL</param>
</port>
<port type="slot" bind="8880">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:sub</param>
</port>
<port type="slot" bind="8881">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:pub</param>
</port>
<port type="slot" bind="8882">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:rad</param>
</port>
<port type="slot" bind="8883">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:hl7sub</param>
</port>
<port type="slot" bind="8884">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:hl7pub</param>
</port>
<port type="slot" bind="8885">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module>xslots.exe</module>
  <param>CERT: SITE:pacs</param>
</port>
<port type="pnp" bind="69">
  <mode>thread</mode>
  <root> C:/Program Files/xService </root>
  <module></module>
  <param>SITE:tftp</param>
</port>
```

</config>

2.2.2 配置说明

项目	选项	说明
type	[http slot pnp]	http: 配置为 http 服务 slot: 配置为 tcp 服务 pnp: 配置为 udp 服务
bind	[1024-65535]	注册端口
mode	[thread process]	thread: 配置为线程调度模式 process: 配置为进程调度模式
root	[absolute path]	xService 运行路径的根目录
module	[path][execute file]	当采用进程调度模式时的子进程可执行文件名
param CERT	[SSL]	xhttps,slots 可选空值或 SSL 选项 SSL 表明采用 TLS3 安全套接字连接
param SITE	user defined	对于 xhttps 服务, 无指定站点值。SITE 由用户资源串指定, 如 http://127.0.0.1:8889/loc/myfile.txt 中, 黑体 loc 即为 SITE。 对于 xslots 和 xpnp, 必须指定值, 如 sub 为 mqtt 的订阅站点, tftp 为 tftp 文件服务站点。

2.3 站点

站点目录位于根目录下, 站点是受调服务例程的入口, 站点用于放置服务例程运行支持库、配置文件、资源文件。

2.3.1 api

xService 运行时所需的动态库均存储于 API 目录下, 受调服务例程的运行文件也位于该目录下。API 的路径需配置于系统环境的 PATH 变量中。

2.3.2 cfg

cfg 目录下放置各服务例程站点的调度配置文件。名称为[sitename].config

举例配置文件	举例配置选项		举例配置说明
oau.config	proc	oau_api[.dll][.so]	oau 服务的受调例程
	path	(\$root)/oau	oau 服务的运行目录为根目录下的 oau 目录
	auth	[true false]	true: 开启 Authorization 鉴权 false: 关闭鉴权

	trace	(\$root)/log/oau	受调例程的日志文件目录
--	-------	------------------	-------------

2.3.3 lic

lic 目录放置用户鉴权访问的认证文件。配置文件样例：

```
<?xml version="1.0" encoding="utf-8" ?>
<config>
<licence>
<CODE>[机构代码]</CODE>
<ORGAINIZATION>[机构名称]</ORGAINIZATION>
<VENDORS>[开发商名称]</VENDORS>
<PUBLIC-KEY>83a13498e0f126a16f2b3084d90478af</PUBLIC-KEY>
<PRIVATE-KEY>925b348b5b7764ebb94329cebfebe606</PRIVATE-KEY>
</licence>
</config>
```

2.3.4 ssl

ssl 目录放置 TLS CA 的认证文件和 RSA KEY 文件。

2.3.5 site

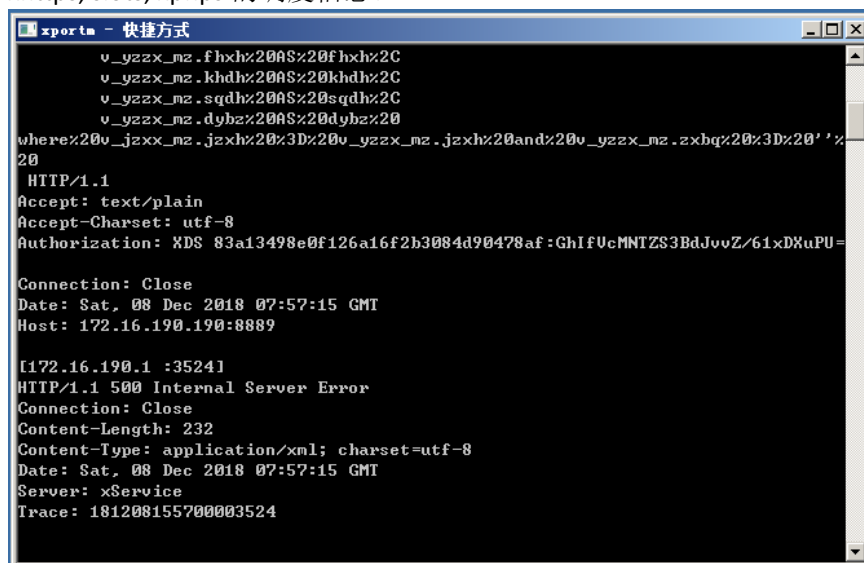
每个受调服务例程都有各自的服务站点目录，以下为常用的站点名说明。

站点名	说明	运行时配置文件
oau	xhttps 授信代理服务	oau.ini
loc	xhttps 共享文件服务	loc.ini
oss	xhttps OSS 访问代理	oss.ini
soa	xhttps soap 服务	soa.ini
pub	mqtt 发布服务	pub.ini
sub	mqtt 订阅服务	sub.ini
rad	radhex 网络服务	rad.ini
dimse	pacs 网络服务	dimse.ini
hl7pub	hl7 express 发布服务	hl7pub.ini
hl7sub	hl7 express 订阅服务	hl7sub.ini
tftp	xnpn 文件服务	tftp.ini
xdb	xhttps 数据库网关	/odbc/[mssql].dsn /oci/[oracle].dsn /mysql/[mysql].dsn

3 监控

3.1 监听服务

监听服务由根目录下的 xhttpm 例程启动，它是一个控制台，用于输出 xhttps, slots, xpnps 的调度信息。



```
xports - 快捷方式
v_yzzx_mz.fhx%20AS%20fhx%2C
v_yzzx_mz.khd%20AS%20khd%2C
v_yzzx_mz.sgd%20AS%20sgd%2C
v_yzzx_mz.dybz%20AS%20dybz%20
where%20v_jzx_mz.jzx%20%3D%20v_yzzx_mz.jzx%20and%20v_yzzx_mz.zxbq%20%3D%20' '%
20
HTTP/1.1
Accept: text/plain
Accept-Charset: utf-8
Authorization: KDS 83a13498e0f126a16f2b3084d90478af:GhIfUcMNTZS3BdJvvZ/61xDXuPU=

Connection: Close
Date: Sat, 08 Dec 2018 07:57:15 GMT
Host: 172.16.190.190:8889

[172.16.190.1 :3524]
HTTP/1.1 500 Internal Server Error
Connection: Close
Content-Length: 232
Content-Type: application/xml; charset=utf-8
Date: Sat, 08 Dec 2018 07:57:15 GMT
Server: xService
Trace: 181208155700003524
```