圣胡安 一 工作原理DNS 基础知识
大西洋标准时间 2018 年 3 月 12 日星期一 一 17:00 至 18:30
ICANN61 | 波多黎各圣胡安

凯西 • 彼得森 (CATHY PETERSEN)： 大家下午好，欢迎参加“工作原理：DNS 基础知识”讲习会。我们邀请了 ICANN 首席技术官办公室研究副总裁迈特 • 拉森 (Matt Larson) 作为我们的演讲人。大家可以看到，宴会厅里已经来了一些人，如果你们不介意继续的话，我们现在开始。迈特？

迈特 • 拉森： 大家下午好。欢迎参加“工作原理：DNS 基础知识”讲习会。人数不多，如果有问题，请举手。我们有充足的时间讲解资料，所以我们可以停下来回答问题。

IP 地址对机器来说很简单，但对人类却很难，这就是我们在这里先讲 DNS 的原因。当我们只有 IPv4 地址时，期望人们可以记住一到两个地址，但这是一个 IPv6 地址的示例，字符数相对较少，这更多地取决于地址中有几个零，记住 v6 地址是几乎不可能的。重点是人们需要使用域名。计算机和路由器使用号码，人们需要域名。

在互联网的早期阶段，域名很简单。我们称之为“单标签域名”，这些域名中间没有点号，他们还不是域名，因为域名还

没有发明出来。早期互联网上的每个域名都对应一个 24 个字符的域名空间。我们将互联网上各个地方每个计算机的每个域名称为“主机名”。主机是计算机的一种花俏的说法。

将这些域名映射到 IP 地址，人们就能使用域名，而之后程序或计算机或路由器等东西也可以使用号码，这就是所谓的“域名解析”。在 DNS 之前，早期互联网的域名解析使用主机文件，叫做“host.text”。你们不必了解这个，这只是历史遗迹。这个文件具有相同的功能，但如果对 Lynx 的 UNIX 很熟悉的话，就会知道它与现代的 Esty 主机文件相比，格式稍有不同。这基本上只是一个文件，一个带有域名和地址的文本文件，因此具有主机名和相应的 IP 地址。

这由一个名叫网络信息中心（简称 NIC）的组织集中维护，他们与美国政府签订了政府合同，以处理早期互联网上的某些网络管理任务，其中一项任务就是维护主机文件。

那是互联网还非常非常小，甚至都不叫互联网，而是叫做 ARPA 网，它是没有美国政府的国防部进行的一次试验，现在我们谈论的是互联网上有数万个主机，所以集中维护这个文件是合理的，更新机制的技术含量也非常低，是通过电子邮件完成的。

当网络管理员将计算机添加到网络、从网络上删除计算机、更改其名称或更改其 IP 地址时，它会向 Nic 发送一封电子邮件，说：“嘿，我做了这些事情，请将这台计算机的 IP 地址改成

这个。”然后 NIC 会维护 host.text 文件的主副本，他们一周释放一个新文件。整个网络中的网络管理员，无论他们何时决定文件过期或可能过期，他们都能下载一个新文件。这不会立即更改，不更新可能也能运行一会儿。你可以通过 FTP 下载。

这是一个技术含量很低的解决方案。如果你仔细想一会儿，就可以预测到一些问题。其中一个问题就是名称争用。如果你用 24 个字符为一台计算机命名，并且网络的规模还在不断增长，那么网络上的设备越多，就会出现更多争用名称的情况，避免重复开始变得越难，事情就越糟糕，这个文件是以非常简单的方式维护的，NIC 在文本编辑器中直接编辑它，背后没有数据库，只有文件和文本编辑器。没有很好的方法来防止重复，重复经常突然就出现了，所以这是一个问题。

另一个很明显的问题是符号化，没有人拥有相同版本的文件，你总是落后。流量和负载是一个问题。在 host.text 使用结束时，文件开始变得非常之大，以至于需要互联网上的大量带宽来下载文件。当时 64 Kbps 的连接速度都很快了，我被告知，这是一段时间之前，但我了解到在 host.text 生命周期快要结束时，下载文件需要更长时间，然后才能更新文件。

换句话说，你永远不可能得到最新的文件，因为等到文件下载完成时，又有一个新文件要下载。需要明确地集中维护这个主机文件，它没有伸缩性，是必须要做的事情。

于是 20 世纪 80 年代初开始讨论 host.text 的替代品，有两个主要目标。一个是为了解决我刚刚讲的扩展问题，另一个是为了简化电子邮件路由，我会介绍一点相关情况。人们在想到 DNS 和 DNS 的动机时，往往只记得第一个扩展问题，但他们可能忘记了或从未意识到电子邮件路由问题，但这也是一个问题。当然，我们在这里的原因是域名系统。

我在一张幻灯片上总结了 DNS。你们将看到这张总结 DNS 的幻灯片。基本上，DNS 是一个分布式数据库，在这个数据库中，数据在本地维护，所以每个人都有自己的数据库部分，他们负责维护自己的数据，但这些数据是全球可用的，因为数据库本身在整个世界、整个互联网中分布。虽然你在本地维护自己的数据，但可以查找并查看其他人的数据。

DNS 遵循客户端服务器模型。解析器是客户端，如果要记住一件关于解析器的事情，那就是他们会发送查询。域名服务器是服务器端，如果要记住一件关于域名服务器的事情，那就是他们会回答查询。解析器发送域名服务器回答的查询。

有一些优化，DNS 使用缓存来提高性能，我的意思是，因为我们谈论的是分布在整个世界的数据库，光速只能那么快，所以当你查询这个数据库时，可能需要进行几次查询，跨越整个世界才能回来，这需要时间。这非常有帮助，事实上，不仅要记住查询的最终结果，而且还要记住所有中间结果，记住它们并缓存它们，下一次可加快此过程，这很重要。

DNS 还使用复制来提供冗余和负载分配。我的意思是，我说每个人都维护他们自己的本地数据副本，但涉及复制，他们不只有一个副本，他们有多个数据副本，并提供冗余。换句话说，如果你只有一个数据副本，发生事情时就没有人可以查找东西，但如果你有多个副本，那么就有冗余。它还会分散负载。如果很多人都在进行查找并且你有多个副本，那么查询负载可以分布在多个副本中。

以下是组成 DNS 的各种组件，在一张幻灯片的照片中标出了所有部分。我们将在本次讲习会中详细讨论这些内容，但我认为这有助于一次性展示这些内容，让大家知道我们的目的。让我们从左下角开始，一张一张地讲。在左下角有一台连接到互联网的设备，这里是一部手机，但实际上可以是连接到互联网的任何设备，需要将域名转换为地址，换句话说，任何使用 DNS 的设备，它都将有一个简单的 DNS 客户端，称为“存根解析器”。

存根解析器的工作是成为应用程序之间的桥梁，比如在本例中，有一个 Web 浏览器图标，存根解析器的作用是成为应用程序之间的桥梁，比如浏览器和整个 DNS 的其余部分。存根解析器接受应用程序请求，例如将一个域名转换为一个地址，然后将其转为它发送的 DNS 查询，并将其发送到称为“递归解析器”的东西。

存根解析器非常简单，它只知道如何做才能接受应用程序的请求，将其转换为 DNS 查询，将该查询发送到递归解析器，然后等待响应。另一方面，递归解析器相当复杂。它知道如何联系各种我们所谓的“权威名称服务器”，DNS 中的数据存储在这里并可供查找。

递归解析器可能需要联系多个权威服务器才能找到答案，它可能会联系一个，权威服务器可能会说，“不，我没有你要找的最终答案，但我可以把你送交给另一个域名服务器。”然后它会联系另一个域名，那个会说：“我没有答案，但我可以把你送交到更近的地方。”

因此，递归解析器足够聪明，可以浏览这些权威服务器并找到答案。如果我们仔细了解递归解析器，就会发现它实际上由一个域名服务器和一个解析器组成。请记住，域名服务器回答查询，所以递归解析器的域名服务器组件会回答来自存根解析器的查询，但接着解析器部分，记住解析器发送查询，解析器部分会将查询发送到权威名称服务器。

然后，我还在展示了缓存，解析器收到的所有内容，从权威服务器收到的每个响应都会被放入缓存中，以便用它来回答未来的查询。这就是 DNS 的大致生态系统。

我想在这里定义一些重要的 DNS 术语和概念。第一个是我们所说的“域名空间”。我说过 DNS 是分布式数据库，那个数据库的结构就是我们所说的域名空间。当我说到数据库时，如

果你对关系数据库比较熟悉，可能会想到它，关系数据库的结构是你有多张表并且表中有行。在每一行中有列和信息。这就是关系数据库的结构。

DNS 数据，域名空间的结构完全不同，这就是我们所说的“倒置树”。我在一张幻灯片上列举了很小一部分 DNS 域名空间的例子。在一个倒置树上，根在顶部，树枝向下生长。

这是一个计算机科学家的树，你不应该对此感到惊讶，计算机科学家的树是倒置的，根在顶部，分支向下延伸。这棵树中的每个节点，所以每张幻灯片上的每个方框，每个节点，都有一个名称，有一个标签。根节点很特殊。树最顶端的根节点实际上没有标签，或者它的标签是“空标签”，标签中什么都没有。有时，你会看到它表示为像我在幻灯片中用引号标示的点号，这只是为了表明没有任何内容。这是域名空间。

我们经常在域名空间中引用这些节点，它们的位置相对于根。根在顶部，然后在根的下面，看看左边，有我们所说的“顶级节点”，它们就在下面。低于顶级的是二级节点，以此类推，沿着树向下移动，贯穿整个域名空间。有时，我们会用像父与子这样的家庭术语来指代节点。

在这个示例中，根是 .com 父，.com 是示例的子，而示例是 .com 的子，可以说是父子关系。这些标签中的每一个都有一组可以使用的有限字符，对于字母、数字和连字符，我们称之为“LTH”，这些字符是 DNS 标签域名中唯一合法的字符，

一个标签的最大长度为 63 个字符。另一个需要了解的重点是，标签域名不区分大小写，可以混合使用大写和小写字母，它们是等效的。

这些节点中的每一个都有一个域名，域名的用途是告诉你节点在域名空间中的位置。域名的定义非常简单，从一个节点开始，沿着相应的路径向上朝根的方向前进，然后写下标签，并在中间插入一个点。大家可以看看底部这里突出显示的注释，我们先写 www，加一个点，然后到它的父，这是例子，我们把例子放进去，再加一个点，然后写 com 和一个点，这就是根。

有一种特殊的域名，就是我们所说的“完全合格的域名”，这种域名与其他任何域名都没有关系。完全合格域名会明确告诉你，该节点在域名空间和 FQDN 中的位置，完全合格域名以点号结尾，而点号实际上是 TLD 之间的分隔符，在本例中为 .come，这就是根。你知道了 TLD，加一个点，然后加上根标签，但根没有标签，根标签是空标签，这意味着域名以点号结尾。

如果所有这些看起来有点熟悉，那么你可能对 Unix 文件系统或 Windows 文件系统很熟悉，这是一个数据结构的例子，也可以表示为倒置树。在一个文件系统中，节点代表文件或目录，你拥有的不是域名，而是一个路径名，文件路径名会告

诉你文件或目录在文件系统中的位置，就像域名告诉你节点在域名空间中的位置一样。

这里有另一个重要的术语，“域”。域的定义非常简单，它只是域名空间中的一个节点，所有内容都在它下面。例如，我突出显示了 domain.com，.com 将是 node.com，所有内容都在它下面。我在这里列出了三个 .com 的域名，实际上有 1.31 亿个，显然幻灯片遗漏了一些。.com 域非常庞大，一切都在 .com 之下，任何以 .com 结尾的域名都在 .com 域中。

把它与术语“区域”对比，这是一个非常重要的术语，因为它包括你所听到的所有东西，理解这一点非常重要。记住，我们有 DNS 的原因是我们需要分布式管理，关于主机名和 IP 地址的集中维护信息没有伸缩性，所以我们必须采用分布式，让每个人来维护自己关于主机名和地址的信息。

域名空间因此被分割，以允许分布式管理和这些分区，这些管理分区被称为“区域”。每个人都有自己的区域，就像一个可以在其中玩耍的小沙盒。他们可以在区域内进行所有更改，而不会影响其他人，他们负责各自的区域并维护它。

区域通过授权创建。域名空间中较高的级别授权给较低的级别。我们将授权区域称为“父”，将被创建的区域称为“子”，此过程始于根，并一直向下。

我举个例子。再说到这个域名空间，如果只看这样的域名空间，我们没有足够的信息来了解区域边界在哪里。只看这个域名空间，我们不知道它如何为了管理目的而分割开。

我给了一个例子，想象一下你正在卫星或空间站上往下看，看到了北美。如果只看北美时，你不能说现实中有三个国家，有加拿大、美国和墨西哥，因为你不知道。

你整天看着北美，但从来不知道管理边界、政治边界在哪里，因为他们没有出现，你必须了解额外的信息。这与区域是一样的。你可以查看域名空间，但是除非你知道授权发生在哪里，否则你不会知道区域边界在哪里。

我来画一些区域边界。我碰巧知道域名空间中授权的位置，这就是为什么我知道这些区域边界的原因。在域名空间的最顶端有根区，然后根区授权给顶级区，可以这样想，一个区域拥有创建或指向它授权的区域的授权信息，这个授权流程会沿着域名空间继续向下。

在这个示例中，根区授权给 .com，然后 .com 区域授权给本例中的 example.com 和 bar.com 以及 foo.com。如果要考虑这些区域的相对大小，就要从根区开始。上次我查了一下，有 1543 个顶级域名区，所以根区相对较小，只需要了解其下 1543 个区域中的授权信息就可以了。

向下看看 .com，.com 的另一方面，如果你们看了在你登记时包里 Verisign 提供的材料，他们的最新报告显示，.com 中有大约 1.31 亿个域名。com 区域非常庞大，它是最大的区域，.com 区域中有很多信息，授权给 1.31 亿个 .com 区域，其下的二级区。授权在二级之下可以继续，我在这个幻灯片上没有距离，但它当然可以，授权可以任意深入到域名空间。

你们记得我刚才说过，DNS 使用复制来实现冗余并提高性能，我们现在讲的就是复制。记住，域名服务器回答查询，如果它完全了解该区域，就可以说域名服务器对区域具有权威性。

也就是权威名称服务器，它知道区域中的内容，如果有人问起，它可以明确地回答说，“哦，是的，你问了区域中的某些东西，相关信息在这里。”或者可以说，“你问了些事情，这个域名不存在，它不在区域中，所以我可以说它不存在。”

区域应该有多个授权服务器，这就是复制，正如我说的，它提供冗余并分散负载。对于每个区域，必须至少有一个权威名称服务器，实际上它将有多个，最好至少有两台权威服务器，因为如果只有一台权威服务器，发生事情时就没有人可以查询你的区域了。

如果你要有多个授权服务器，就必须保持其上的信息同步。也就是说所有权威服务器上关于一个区域的信息应该是相同的。如何才能做到呢？好消息是 DNS 协议内置有一种方法做到。

跨权威服务器同步区域数据的功能是内置的，并且有一个称为“区域转移”的流程，可让你移动区域数据。你可以将一个权威服务器授权为“主”服务器，在这里你可以对区域进行更改，然后将其他权威服务器作为“辅助或从属”服务器，这些权威服务器从主服务器加载区域数据，联系主服务器，进行所谓的区域转移，然后将区域从主服务器复制到辅助服务器。

值得注意的是，一个区域的所有权威服务器都是平等的，它们都拥有相同的数据。主服务器和辅助服务器的唯一区别是他们从哪里获得数据？主服务器将区域数据从其磁盘上取下来，辅助服务器从主服务器加载区域数据，但主服务器和辅助服务器上的数据完全相同。

当然，他们可能会短暂不同步，因为在主服务器上进行更改后，将生成更多最新信息，但随后会传播到辅助服务器，再同步。这样很好，是内置在 DNS 协议中，你不必担心要自己保持域名服务器同步，它会自动同步。

现在我想向下移一层。我们已经讨论了区域，现在来看看区域内部，讨论区域中的数据。记住，域名空间中的每个节点，我展示的图表上的每个方框都有一个与其相对应的域名，可以这样想，给定的域名可以具有不同类型的信息，与之相关的不同类型的数据。

最常见的是 IP 地址。我们可以将一个 IP 地址与一个域名相关联。我们将域名附带的这种信息称为“资源记录”。资源记录

是 DNS 中的数据，资源记录有不同的类型，以存储不同类型的数据。最常见的资源记录类型用于存储 IP 地址。IPv 和 IPv6 是不同类型的记录，但它们是存储其他数据类型的资源记录。

一个区域仅包含一堆资源记录，区域的所有资源记录都放在一个文件中，我们称之为“区域文件”。每个区域都有一个区域文件，绝不能混合一个文件中多个区域的记录。区域只不过是其资源记录的集合。

我来向大家展示这些资源记录的样子。其实有一种方式把它们写下来，用文本的标准方式把它们写下来。资源记录有五个字段，我们不需要全部了解。需要了解的是，它们是不同类型的资源记录，它们携带相应类型的数据。这些是最常见的资源记录类型。

我提到过，我们有一种针对 IPv4 地址的记录类型，称为“A 记录或地址记录”，我们还有一种存储 IPv6 地址的记录类型，称为“AAAA 记录”。然后，我们将在这里简单讨论一下其他几个类型。这个列表列出了最常见的类型，但实际上有很多其他类型的资源记录。

我上一次查看的时候有 84 种不同的类型，并且有一个叫 IANA 注册管理机构，我在幻灯片上写了，我不想把 URL 读出来，你们可以去那个网页看看是什么样的。根据资源记录字段的大小，最多可以有 65,000 个，而我们远远没有达到，只有 84 个。

如果你想在 DNS 中存储一个新东西，可以去 IFT，写一个互联网草案，让人们相信你的想法应该成为一个新的 DNS 类型，你可以创建它并在这个注册管理机构中获得一个类型，然后就能拥有新的东西，你可以把它放到 DNS 中，人们都是这么做的。并不经常这样，我们只有 84 个，但人们确实有他们想要放入 DNS 的新东西，他们想创建一种新类型来存储新的数据类型。但到目前为止，DNS 中最常见的数据类型就是地址。

DNS 最常见的用途是将域名映射到地址，我在这里展示了两种地址类型的记录。下面是一个地址记录和一个 AAAA 记录的实际文本表示示例。左边有一个域名，example.com，然后有类型，第一个是用于地址的 A 记录，还有实际地址。

这是一个资源记录的例子，它在 example.com 区域的区域文件中，此资源记录仅表示，example.com 具有此 IP 地址。下面是 AAAA 记录，表示 example.com 具有此 IPv6 地址。大多数 DNS 都由 A 和 AAAA 记录组成，因为我一直在说，这就是 DNS 的主要目的，即将域名映射到 IP 地址。

还有其他类型，我认为有趣的是，大多数这些类型都被消费 DNS 信息的人所使用，他们在 DNS 中查找信息，因为他们需要进行诸如连接到 Web 服务器之类的操作，因此他们的 Web 浏览器需要查找域名来解决映射问题。

但是，只有部分类型被使用，大部分是由 DNS 本身使用的，主要有 NS 记录和 SOA 记录，我们将非常简单地讲讲这些

记录，这些类型对是否接受 DNS 本身一点儿也不在乎。有趣的是，这些类型与其他类型类似，我喜欢把它看作仓库，如果可以将其与仓库相比的话。

假设你租了一个仓库，并且你有一堆东西想要放到仓库里。为了使用仓库，你不仅要将卡车倒过去，开始将箱子搬进仓库，还需要搭建一些货架，架起货架后，就可以把箱子、你在意的物品、箱子里的货物放到货架上。

没有货架的仓库一点也不好，DNS 也是一样，你可以把这些 NS 记录 和 SOA 记录看成货架，它们必须存在才能使 DNS 正常工作，但 DNS 之外没有人真正关心这些，他们只关心所有其他类型，如 A 和 AAAA。

我来讲一下这些 NA 记录。这些是对一个区域的权威名称服务器的表述。这里的示例显示了两个 NS 记录，这些 NS 记录表示 example.com，该区域有两个权威名称服务器，一个叫做 NS1.EXAMPLE.COM，一个叫做 NS2.EXAMPLE.COM。左边是区域的名称，右边是域名服务器的名称。

现在，NS 记录变得有点复杂了，因为它们实际上会出现在两个地方。它们会出现在父区，也会出现在子区。在这里的方框中，有 .com 区域的 NS 记录列表，.com 区域实际上有 13 个权威名称服务器，看右边，它们被命名为 A.gTLD- SERVERS.NET 到 M.gTLD-SERVERS.NET，这个 NS 记录列表有 13 条记录，它们会出现在两个地方。

我把这里放大一点。它们会出现在根区，在本例中，根区是父区，这些父区中的这些 NS 记录实际上就是授权，告诉 DNS 的其余部分，根区下是 .com 区域，然后在这个 .com 的例子中，NS 记录列表也会再次出现在所命名的区域中。.com 的 NS 记录出现在根区、父区中，然后出现在 .com 本身中。

当我们讲到如何在 DNS 中查找数据时，你会发现这些 NS 记录出现在父项中有多么重要，因为我会提前说到 DNS 中的解析方式，在 DNS 中查找内容的方式是，从根开始，然后跟随这些授权指示，跟随 NS 记录查找。如果你要查找 .com 下的东西，可以从根开始，在根区中，会看到 .com 的授权，然后可以进入 .com 域名服务器，在下面可以找到一个授权等等，但马上会看到更多内容。

NS 记录只包括域名，看看我给出的例子，example.com，其中一个域名服务器是 NS1.EXAMPLE.COM，但只有这一个是不够的，因为你之后需要 NS1.EXAMPLE.COM 的 IP 地址，如果你真的要联系它的话。在某些情况下，授权信息也需要包括地址记录，我们称之为“粘合记录”。如果你们听别人讲过粘合记录，它是域名服务器的的地址记录。

在每个区域中，还有另一个记录称为“SOA 记录”，我不想详细讨论这个记录，我只想指出它存在。每个区域都有的一个 SOA 记录中位于顶端，我们称之为区域的“顶点”。这里的大

多数值都与我之前提到的区域转移有关。它们告诉权威服务器如何同步、多久同步一次区域。

我们回到 DNS 的第二个目标，DNS 要解决的第二个问题是帮助电子邮件路由。DNS 必须解决的问题是，如何根据电子邮件地址发送邮件？早期，在 DNS 之前，电子邮件地址一般是用户@主机名，它是其中一个最多包含 24 个字符的域名。

那是你的电子邮件地址，这意味着在 DNS 之前，你的电子邮件会发送到电子邮件地址右侧的主机名。不能说，我的电子邮件地址是 MATT@FOO，但是你应该将邮件发送到名为 Bar 的机器或其他地方。不，如果你的电子邮件地址是 MATT@FOO，那么你的邮件就会被发送到名为 FOO 的机器上，这就是必须了解的地方。

DNS 的其中一个目标是变成十倍，可以说“这是我的电子邮件地址，这里是邮件应该到达的地方，你应该把它发送到别的地方。”DNS 提供这种灵活性。有一个叫做邮件交换记录的记录，告诉你一个域的电子邮件应该发送到哪里，这里有一个例子。example.com 域名的这些 MX 记录说明邮件应该发送到哪里。

对于所有用户名@example.com，这些 MX 记录说明应该将其发送给名为 MAIL.EXAMPLE.COM 的机器。这是首选值，叫做 10 号和 20 号，这是不太直观，因为首选值越低，邮件服务器更理想。这两个 MX 记录说明，对于发送到 example.com 上任

何一个地方的电子邮件地址，你应该将其发送到 MAIL.EXAMPLE.COM，但如果由于某种原因不能，那么应该尝试 MAIL-BACKUP.EXAMPLE.COM。

任何要发送邮件的邮件服务器都必须能够查找域名的 MX 记录。DNS 和基于 SMTP 的电子邮件之间有非常紧密的耦合。当邮件服务器收到要发送的邮件时，它会查找电子邮件地址的 MX 记录，这就是它知道如何发送邮件的方式。

到目前为止，我们一直在谈论将域名映射到 IP 地址，这是一项重要任务。有时你想要的相反。我们将域名到 IP 称为“正向映射”，但是如果你想从 IP 开始并知道其域名是什么，该怎么办？有时候你想要这样做。

想象一下，例如，有一个名为 Trace Route 的网络故障诊断工具，可以显示你和你想要到达的 IP 地址之间的每个路由器。当显示每个路由器的 IP 地址时，你可能对 IP 地址感兴趣，也可能只想知道该 IP 地址在哪里？谁操作它？它叫什么名字？这是反向映射的一个例子，它要获取 IP 地址并找到相应的域名。

回想一下主机表。如果只有一个包含域名和 IP 地址列表的文件，并且想进行正向映射，那么非常简单，你有一个名称，想要将其映射到一个 IP，你可以浏览该文件，直到找到域名就可以了。如果想进行反向映射怎么办？这很简单。你可以浏览

该文件，直到找到 IP，然后就知道域名了。使用主机表非常好。使用 DNS 怎样做呢？

我们回到域名空间的例子。域名空间的构建方式使其非常易于查找域名，但在本例中，无法查找 IP 地址。如果我想查找一个 IP 地址，比如说 WWW.EXAMPLE.COM，大家可以看到我从根开始，向下一直到我找到 WWW。但如果我从 IP 地址开始，该怎么办？答案是就我展示的来看，无法在 DNS 中这样做，因为无法查找 IP 地址。

我们必须有办法将 IP 地址转换为域名，这样才能作为域名进行查找，所以这实际上就是我们拥有的。还有另外一种记录叫做“PTR”，说明 IP 地址进入 PTR 记录所在的特殊域名，这可以让你以域名查找 IP 地址，然后找出它们对应的名称。对于 internet address.arpa，IPv4 地址位于名为“IN-ADDER”的域名下，然后 IPv6 地址位于名为 IP6.ARPA 的域名下。

我举个例子。这是域名空间树。我只展示新的部分，大家可能没有意识到这一部分的存在。右边有 EXAMPLE.COM，我们知道了这个，但还有 INADDER.ARPA 域，在本例中，看看底部，这就是 EXAMPLE.COM 区域的 PTR 记录。大家在右边可以看到一个地址记录，说 EXAMPLE.COM 的 IP 地址是 192.0.2.7，如果你想查找 EXAMPLE.COM，就可以在 EXAMPLE.COM 看到地址记录，并知道地址。

我怎样才能知道与 IP 地址 192.0.2.7 相对应的域名是什么？你必须把这个 IP 地址转换成一个域名，可以把 IP 地址翻转过来，为其添加 IN-ADDER.ARPA，然后查找 PTR 记录。这样做的唯一理由是让每个人都了解规则。每个人都明白我在说什么。如果你有分配给你的 IP 地址空间，地区互联网注册管理机构，简称 RIR，他们合作管理 IN-ADDER.ARPA 域，并且你可以获得与授权给你的 IP 地址相对应的区域。

举个例子，已经分配给你 192.0.2/24，所有以 192.0.2 开头的地址，你将拥有 2.0.1。192.INADDER.ARPA 域分配给你，如果你希望人们能够对你的 IP 地址进行反向映射，你就必须把 PTR 记录放在里面。这有点不合适，但行得通。

我认为大多数人会同意反向映射不如正向映射那么重要。正向映射就是让你在浏览器中输入域名，然后进入网站。反向映射往往更多地用于诊断和故障排除。除了网络工程师或系统管理员之外，正常人可能不太关心反向映射。

正如我所说的，资源记录类型有很多种，下面这个例子更多的只是为了让你们了解人们想到的可以放入 DNS 中的其他类型的数据。这里还有一个例子，对于非常小的区域来说，区域文件可能是什么样的。

这是我们假设的 EXAMPLE.COM 区域的一个区域文件，我知道我没有详细介绍每个记录，但这是一个类似于互联网上大多数区域的小区域，因为如果要考虑它，对于互联网上的大多数域

名，你可能要做两件事。你要有一台 Web 服务器，有一个网站，并且你想接收电子邮件。

现在，显然域中有更多其他内容，各种各样的名称，但很多域名，例如我的个人域名，我用于电子邮件，我关心电子邮件，我有一个小网站。我的域名的个人区域看起来像这样，这就是你需要支持这些应用程序的全部内容。

我们有一个 EXAMPLE.COM 的IP地址，这是我们的 Web 服务器的地址。你们可以看看这个区域文件，可以猜测 192.0.2.7 是 EXAMPLE.COM 网站的 IP 地址，因为我们有一条显示将 EXAMPLE.COM 映射到 IP，然后你会看到一些 MS 记录，说明 EXAMPLE.COM 将邮件发送到哪里。

现在我想谈谈解析过程。这是你在 DNS 中查找内容的方式。我在讲习会开始时那张图片上展示的 DNS 组件，存根解析器、递归解析器和权威名称服务器，他们共同在域名空间中查找数据。

值得注意的是，一个 DNS 查询总是有三个参数，它有一个域名，如 WWW.EXAMPLE.COM，它有一个不要查找的数据类型，在本例中是用于地址的 A 记录，还有这个称为“类”的值，我跳过了，没有讲。类是人们早期认为他们可以用来将 DNS 扩展到其他类型的网络的方式，但实际上它还没有被使用，但已经被纳入到了 DNS 中，我们一直是这样做的。

在这种情况下，类总是被称为“互联网类”，你永远不必担心。在这种情况下，它确实只是域名和重要的类型。如果你要进行 DNS 查询，问域名服务器一个问题，则必须指定域名和类型。

有两种类型的查询，记住存根解析器就像你的手机、烤箱、冰箱、笔记本电脑等任何连接到互联网的设备中的东西，需要把域名转换成地址或其他信息，所有这些设备都有一个存根解析器。存根解析器发送所谓的递归查询，递归查询是递归解析器的信号，它说：“嘿，我是一个存根解析器，我只需要你给我答案或错误。我不能处理任何东西，也不能接受部分答案，我需要问题的完整答案。”

另一方面，递归解析器比较智能，它们可以接受这些属于转介的部分答案。它们会发送一种查询，指出它们可以在响应中进行转介。正如我说过的，如果你想在 DNS 中查找某些内容，那么就要从根区开始，然后一直向下，跟随授权指示。

权威服务器对根区具有权威性，它们拥有根区中的所有信息，这就是权威的含义，我们称之为“根名称服务器”。如果你要在根区开始解析，就需要能够联系根名称服务器，那么如何才能找到谁是根名称服务器呢？答案是，它们必须进行配置，没有办法发现它们，它们必须在每个递归域名服务器上进行配置。这与其他网络参数不同。

当我的手机在会议中心的 ICANN WIFI 网络上出现时，它使用了一种称为“动态主机配置协议”的协议，简称 DHCP，我的手机问网络：“嘿，我是网络上的新设备，我需要一个 IP 地址。”我的手机对这个网络一无所知，网络说：“好的，这是你的 IP 地址，这是一些你需要知道的其他配置参数，包括要使用的递归域名服务器的 IP 地址。”

这是设备什么都不知道的例子，网络会告诉它需要的所有信息。这不适用于递归名称服务器，你无法在没有配置的情况下打开递归名称服务器，它必须知道根名称服务器是哪个，其 IP 地址是什么。每个递归名称服务器都需要的一个特殊文件，叫做“根提示文件”，该文件拥有根名称服务器的名称和 IP 地址。

好消息是如果你安装了递归名称服务器，比如说在 Lenox 机器上，有人执行了打包工作，当他们打包递归名称服务器软件时，纳入了根提示文件，有些递归解析器甚至还有根服务器的名称和 IP 地址，作为其软件代码的一部分。通过这个 URL 可以获取根提示文件，就是这样的。

有 13 个根名称服务器。有 13 个服务器对根区具有权威性。左边的点表示根区，我们有 13 个 NS 记录，这些是根区的名称服务器的名称，大家可以看到它们称为 A.ROOT-SERVERS.NET 到 M.ROOT.SERVERS.NET，它们拥有这些名称，然后在 IPv4 地址下可以看到，还有 IPv6 地址。每个根名称服务器都有 IPv4 地

址和 IPv6 地址。为了进行解析，递归名称服务器必须拥有此信息，它必须知道根名称服务器的名称和 IP 地址。

我简单讲一下根区以及信息如何进入根区。记住，根区中有什么？有关于顶级域区的信息。有 TLD 的 NS 记录，管理根区是一件复杂的事情。它由两个组织共同管理，ICANN 的角色叫做“IANA 职能运营商”，ICANN 子机构 PTI 处理该角色，然后另一个组织是 Verisign，扮演名为“根区维护人”的角色。

这种安排非常久远了，要追溯到 20 世纪 90 年代早期，Verisign 当时叫做 Network Solutions，Verisign 是在 2000 年提出的。在 ICANN 成立之前，由南加州大学履行 IANA 职能运营商的职责。这是一个非常久远的安排，有点复杂但根区就是这样运作的。

ICANN 和 Verisign 这两个组织合作汇总根区中的数据，制成根区文件，然后我们需要为根区设定权威服务器，根名称服务器。有 12 个不同的组织运营根区的权威名称服务器。这有点不寻常，对大多数区域来说，只有一个组织运营所有权威服务器。

以 .com 为例。我在 Verisign 工作，知道一点相关情况，Verisign 运营着 .com 的所有权威服务器。对于其他区域，许多公司的做法是他们可能自己运营者一些权威服务器，或者将其外包给第三方提供商，或者可能有多个第三方提供商管理冗余，而不是 12 个不同的组织，这很不寻常。

这里有 13 个根名称服务器字母。即使称为 A.ROOT-SERVERS.NET，有人简称为 AROOT 到 MROOT，这些是运营 A 到 M 的组织。这是一个有趣的团体，除了运营根名称服务器之外，他们没什么共同之处。大家看看这个名单，可以看到有商业组织、教育机构、非营利性组织、ISP、美国政府部门，这只是一部分，可以追溯到很久以前，我们 20 年前就在讨论。

这个运营商名单一直没变，20 年来一直如此，还发生一大堆复杂的故事，我们都没有时间去探究。我们有 13 台根服务器和 12 个组织。有 12 个是因为 Verisign 运营两个，他们运营 AROOT 和 JROOT。这是根区、根名称服务器和根运营商。

如果你想了解更多有关根运营商和根服务器的信息，可以访问 ROOT-SERVERS.ORG 访问，了解所有根服务器的位置和一点相关信息。

我想简单介绍一下根区的变更流程。根区中的所有信息都与 TLD 相关。如果 TLD 经理想要进行更改，如为其 TLD 添加权威服务器，为其 TLD 移除权威服务器或更改其中一个服务器的名称或 IP 地址，他们会将更改提交给 IANA 职能运营商，即 ICANN 所有的 PTI，PTI 会执行一些检查，并更新他们拥有的根区数据库，然后将该请求发送给根区维护人，即 Verisign。

Verisign 会执行更多检查，Verisign 会更新其根区数据库，创建根区文件并使其可用，然后 13 台根服务器会下载该文件并

使其可用。这是对该流程非常非常粗略的描述。我只是想说明不同的组织是如何合作来完成这项工作的，但我之后会介绍更多过程。这是为根区的简单介绍。

我们现在来讨论是如何解析的。左下角有一个手机，有人在手机上打开了 Web 浏览器并输入 WWW.EXAMPLE.COM，Web 浏览器调用了根服务器，根服务器是一段非常简单的代码，这里的橙色表示这其实是一个 API 调用，这是 Web 浏览器程序的一部分，叫做存根解析器，另一个程序或程序的一部分有时只是一个函数调用，说："我需要 WWW.EXAMPLE.COM 地址。"

记住，根服务器非常简单，存根解析器只知道一个递归服务器或者多个递归服务器的 IP 地址，即应该将其查询发送到的服务器。存根解析器会将 Web 浏览器的请求转换成 DNS 查询，然后它会将该 DNS 查询发送到经配置的递归名称服务器，在本例中是 4.2.2.2，这是一家名为 Level 3 的 ISP 运营的一个递归名称服务器的真实 IP 地址，这个递归服务器是他们调用打开的，因此任何人都可以向该递归服务器发送查询，我只是用这个 IP 地址举个例子。

存根解析器会要求递归解析器进行解析，让其现在解析 WWW.EXAMPLE.COM 的地址更有意思，我说过记住递归解析器有一个缓存，但要使其更有意思，我们要假设这个递归解析器已经打开了，因此其缓存中什么都没有，只有根服务器的名称

和 IP 地址。递归解析器会挑选一个根服务器，比如选了 LROOT，它会问其从存根解析器收到的相同的问题，即 WWW.EXAMPLE.COM 的 IP 地址是什么？

现在根服务器无法直接回答这个查询，因为根服务器不知道 WWW 的 IP 地址。它不知道关于 EXAMPLE.COM 的任何信息，但他知道关于 .com 的一些信息，因为根区包含 .com 的授权。根服务器会返回我们所谓的转介到 .com，说：“嘿，这是 .com 的域名服务器和 IP 地址。”

此事递归解析器就会缓存该相应，以便未来使用信息，然后进行所谓的“跟随转介”，挑选一个 .com 服务器，向其发送相同查询。实际的 .com 服务器称为 C.gTLD-SERVERS.NET，Verisign 根据根服务器的命名方式为 .com 服务器指定相似的名称。

像 A 到 M ROOT-SERVERS.NET 一样，这些都是 .com 服务器的名称。我们的递归解析器选择了一个 .com 服务器 C.gTLD-SERVERS.NET，问了它从存根解析器收到的相同问题，它问根服务器：“WWW.EXAMPLE.COM 的 IP 地址是什么？”

现在，.com 服务器不知道 WWW.EXAMPLE.COM 的 IP 地址，但它知道谁是 example.com 的权威名称服务器，因此它可以发回该列表，将转介发回给 .com。

我们的递归解析器会将其缓存起来，并跟随该转介，第三次将相同的查询发送给 example.com 的其中一个权威服务器，即 NS1.EXAMPLE.COM。

此时，NS1.EXAMPLE.COM 对 example.com 具有权威性，它知道 WWW.EXAMPLE.COM 的 IP 地址，所以它会将其返回给缓存它的递归解析器，将其发回给存根解析器，即传回给应用程序，现在应用程序知道了 Web 服务器的 IP 地址，就可以联系它、下载网页并进行后续操作。这就是解析过程，非常简单的描述。像 DNS 的大多数东西一样，它会变得越来越复杂，这只是很简单的介绍。重要都是从根开始一路向下。

由于有缓存，因此你不必总是从根开始。我突出显示了我提到的所有缓存。如果有人一来就要访问 FTP.EXAMPLE.COM，该怎么办？

存根解析器会组译 DNS 查询并像以前一样将其发送给递归解析器，但现在既然递归解析器进行了缓存，它知道 .com 服务器，知道 example.com 服务器吗，因此不必从根开始，也不必转到 .com，可以直接转到 example.com 权威服务器，问问题，获得响应，也会缓存答案并返回给存根解析器，从而返回给应用程序。大家可以看到缓存如何极大地提高速度。如果不缓存，互联网上的所有动作都会慢很多。

在最后一张幻灯片上有一个高度概括的连接图片，我们一直在讨论了 DNS，但考虑到 ICANN 背景下的域名时，我们就想到

了权威 DNS，想到了包括注册人、注册服务机构和注册管理机构的更大的图片。

我展示这张更大的图片只是想告诉大家域名世界的其他参与者在哪里。注册人一般通过网站与注册服务机构沟通，购买域名或进行变更。然后注册服务机构会与注册管理机构沟通。注册管理机构的主要组成部分是数据库，用于保存注册的域名及其相关信息。注册管理机构必须使数据库中的信息与 DNS 相关联，使其在权威名称服务器上可用，这些就是递归解析器查询的内容。希望这对理解这张更大的图片有所帮助。

这就是我要讲的所有幻灯片。我们还有一点时间。如果大家有问题的话，我乐意回答任何提问。

凯西 • 彼得森： 提醒一下，提问时请报上自己的姓名和所属机构，谢谢。

迈特 • 拉森： Adobe 会议室中有要提问的吗？好的。好的，大家都饿了。其实我也有点饿了。

玛丽莎 • 理查兹 (MALISA RICHARDS)： 其实，我有一个问题要问。我是玛丽莎 • 理查兹，来自 [听不清] 的学员。安装根区服务器时采用什么标准？

迈特 • 拉森： 不同的运营商有不同的标准，每个运营商都有自己的政策。我没有讲这个，但在这个地方所有根服务器都是任播的，也就是说指定的根服务器 IP 地址不只有一个服务器，而是有多个服务器，任播技术采用互联网的底层路由系统，允许网络中多个位置的具有相同 IP 地址的服务器应答。

所有运营商都采用任播技术以拥有多个实例，我们用特殊的根服务器字母称呼它们，并且不同的运营商有不同的政策。我知道对于 ICANN，对于 LROOT，我们很高兴能运营 LROOT 实例，它只是一个服务器，我们的要求是你可以买服务器，将它放到网络中，供应机架空间、电源和连接、带宽，然后我们运营它，作为 LROOT 的其余部分运营。不同的根运营商有不同的政策。

玛丽莎 • 理查兹： 还有一个问题。我看了你推荐的网站，发现与南美和北美地区相比，特别是加勒比海地区，并没有很多根服务器。你能解释一下为什么会出现这样的情况吗？

迈特 • 拉森： 我不知道其他运营商是怎样的，可能那些不同网络的运营商没有与根运营商接洽，尝试在那里安装根服务器。至少对 ICANN 而言，标准非常低，如果你想买一个服务器，我们很乐意在你的网络中部署一个。谢谢大家。噢，好的，请讲。

诺曼 • 沃普特
(NORMAN WARPUT)： 我是来自 Vanuatu 的诺曼 • 沃普特，ICANN 英才计划学员。感谢你的介绍。我有一个关于根区变更流程的问题。这适用于 ccTLD 顶级域吗？

迈特 • 拉森： 我不是很明白你的问题。

诺曼 • 沃普特： 更改根区的流程适用于国家和地区顶级域吗？

迈特 • 拉森： 是的，当然。从 DNS 的角度来说，TLD 就是 TLD。我们把 TLD 分成了 ccTLD 和 gTLD 等，还有赞助类和非赞助类 gTLD，但这些都是我们对一些事务的额外分类。从 DNS 的角度来看，TLD 就是 TLD，不管怎么叫，无论是 ccTLD、gTLD 或其他东西。还有其他问题吗？

安德鲁 • 弗雷泽
(ANDREW FRASER)： 相关说明在哪里，你将要去哪里或者如何指向递归服务器？在哪个级别上或通过 ISP 吗？因为递归服务器有多种选择。

迈特 • 拉森： 对的，最后在每台设备上都会配置，当你的设备连接到网络时，网络就会告诉你。除了获取 IP 地址时，每个网络都会告诉你递归服务器的 IP 地址。然后运营网络的任何人都必须拥有递归服务器。

安德鲁 • 弗雷泽： 所以，我的手机会根据我的提供商预先配置？

迈特 • 拉森： 我不会是预先配置，当它连接到提供商的网络时，你的提供商会告诉它：“这是 IP 地址，这是一些其他配置，包括要使用的递归服务器。”也就是他们运营的服务器。但是如果你想，你可以覆盖该服务器，选择另一个。即我们所谓的第三方递归 DNS 提供商，有时叫做公共 DNS 提供商或开放解析器，有很多名字，这些公司运营任何人都可以使用的递归服务器。

我知道这样做的第一家公司是 Open DNS，我记得当他们这样做时，我的反应跟许多人一样，“为什么要这样做呢？”递归服务就好像网络服务，为什么我要把我依赖在网络上进行操作的事情看得如此重要，为什么我要依赖网络之外的域名服务器，Open DNS 的价值主张是，我们可以提供额外的服务，我们可以根据名称进行内容过滤。

你可以告诉我们，我不想解析与赌博或成人内容等相关的名称，而且他们也可以做类似的事情，我们不会解析我们知道是

托管恶意软件的网站或从安全角度来看是不好的域名。我记得 Open DNS 认为这是一个荒谬的想法，但随后 Cisco 以 6 亿美元收购了 Open DNS，我猜他们会笑到最后。但在 Open DNS 之后，例如 Google Public DNS，Google 预定的目标是提供任何人都可以使用且采用 DNSSEC 验证的非常可靠的递归服务。

我不是很了解 DNSSEC，但这会给 DNS 添加加密验证。Google 非常认同 DNSSEC，他们说：“我们提供的这个服务可以免费使用，而且你将获得额外的 DNSSEC 保护。”还有其他公共 DNS 服务，Verisign 有一个，就是 Quad 9。如果你愿意，可以覆盖你的设备配置，使用那些。

还有其他问题吗？好的，谢谢各位。

凯西 • 彼得森： 谢谢大家。谢谢迈特。谢谢我们的速记员和口译员以及技术团队，工作非常出色，再次感谢。

[会议记录结束]