加州大学钱志云：那些计算机应用领域的脑洞是怎么产生的

钱志云　网安国际　今天

原文标题为《如何在计算机应用领域寻找研究想法？》

多年以来，我指导并与十几位博士生合作，在这个过程中发现了一些常见的问题，这些问题阻碍了他们在研究中取得更好的进展。作为一名博士生导师，在本文中我将尝试给予计算机科学领域的学生一些指导建议。我的研究领域是网络安全，因此从事这一特定领域的学生可能会发现本文所给出的事例与他们更为相关（但也应该推广到其他实用性的领域，比如网络、系统和体系结构方向）。本文的观点大多是基于个人经验，因此可能会有失偏颇。尽管如此，我还是希望它能够对一些研究人员有用。对于那些在攻读博士学位之前从未做过任何研究的人来说，可能考虑从哪里入手研究都令人害怕。关于这一点存在两种情况：

> 你有幸（或不幸）身处在一个工作超级高效的团队，有很多想法向你抛来——当你与青年教师一起工作时，这种情况经常发生（当然也有例外）。
>
> 你身处在一个研究体系非常成熟的团队，你的导师不再给学生提供具体的研究思路。相反，他们可能会做的是给出一个非常高水平的方向，连同几篇相关的论文。这就是你脑洞出一个研究思路所能用到的全部内容。

如果你属于第一类，那么你的第一个研究项目从获得、推进到完成的整个过程显然会是令人满意的，除非你并没有真正地从头至尾完成整个研究过程。事实上，在我看来，直接接受别人提供的研究想法是在逃避一个研究项目中最困难的一步——想出一个好的研究思路。凡事都有取舍，你确实能够更早地发表论文，而且可能根据你的导师（或者博士后亦或高年级研究生）选定的思路发表了一篇非常优秀的论文。但是，你失去了训练自己独立寻找研究思路的机会，而这是一个博士所必备的重要技能。作为一个导师，我也会觉得内疚，例如，我完成了发现漏洞过程中的困难部分，然后让学生执行其余内容。

如果你属于第二种类型，则可能是成败参半的情况。要么你想出了一个好思路去执行，要么浪费了攻读博士学位生涯的头一两年，意识到这个思路没有任何用处然后决定放弃。我读博士的时候，目睹了第二种类型的博士生大约有一半选择退学。在上述任何一种类型的团队中（或介于两者之间的类型），你都应该尽早开始训练自己寻找研究思路的能力。否则，我认为这会是一个很大的缺陷，并可能在未来毕业后对你造成很大的影响，例如，作为学术界的教授或工业界的研究员，在独立领导研究项目时你可能会进行得很艰难。

下面是我觉得很有用的一些小贴士。

学会阅读论文，培养自己的兴趣

思路不是突然冒出来的。产生新思路最常见的方法之一是阅读其他论文并获得灵感。需要特别关注相关的研究课程，在那里你将开始阅读大量的论文。同时，你也需要写论文阅读笔记来表达你的观点、意见和任何有建设性的想法。最初，每周每节课读3~4篇论文是有压力的（我还是学生时在同一学期上过两节这样的课），但要坚持住。一个常见的误区是：你必须理解一篇论文的全部技术细节，才算完成论文阅读。不，那既不是主要目标，也不是在有效利用你的时间！在这样的课程中，欣赏和批判研究想法是一个学习的过程，例如，一篇论文为什么是好的或坏的？是什么使得一篇论文吸引人？你不必阅读论文的每一处细节来回答这些问题。关于如何阅读论文，其实有很多很有帮助的文章。例如，S.Keshav写的《如何阅读论文》。

更重要的是，找到你最感兴趣的论文类型，多问问自己为什么。以网络安全领域为例，其涵盖的子领域十分广泛，甚至我都可能无法掌握所有研究内容。在研究领域上，任何计算机科学领域，比如操作系统、网络协议，或者任何软硬件，都有其相应的“游戏规则”和“安全威胁”可以被研究。在网络安全中，人类往往是最薄弱的一环，但同时也扮演着重要的角色。就研究风格而言，网络安全的涉及范围从新型攻击和利用技术，对新兴系统或算法的分析，到防御方案、测量分析等等。另一个维度是一篇论文中所使用的研究方法：人工分析、逆向工程、程序分析、形式化方法、基于硬件的系统设计、数据驱动方法，如机器学习和人工智能。找到你喜欢的论文类型将有助于培养你的研究兴趣，最终缩小研究范围并形成一个新的研究思路。

在我职业生涯的早期，我沉迷于可以攻破最新防御体系的新型攻击技术。这是创造性的、优雅的、极度让人满足的，并能够发现其他人看不到的安全缺陷。当我读到这样的论文时，我总是问自己：“这些人是如何找到缺陷的？开展这样的研究需要什么技能？”。这种研究风格，在有意或无意间引导我培养出了攻读博士学位所必需的思维定势和各种技能。由于我的导师当时的研究专注点在于网络，所以我很自然地致力于研究网络领域中的安全问题（如TCP安全）。然而，在当时我并没有机会掌握诸如程序分析、形式化方法和机器学习这类过硬的技术。当我即将毕业时，我的研究兴趣开始发生转变，因为我意识到，在解决安全问题方面，如果没有适当的自动化技术，是不可能真正做到可持续或可扩展的。因此，我开始学习程序分析、关于模型检查和机器学习/人工智能方面的知识。这对我个

人来说非常有帮助。当然，每个人情况各异，你总能找到自己的兴趣，并以此发展出一条独特的研究之路。其实，这正是学术自由之美！

认清产生研究思路的范式

每一篇论文背后都有一个独特的故事，但它们产生的方式往往可以归类为几种范式。其中大部分是我从研究经验中提炼出来的，也有一些是从其他研究者那里借鉴得来的。例如，来自加州大学圣地亚哥分校（UCSD）大学的Philip Guo教授有一篇很棒的关于各种研究范式的博客文章，这些范式引起了我的很多共鸣。以下是我所提供的版本，并通过具体的例子加以说明。注意这些范式可能并非完全契合。它们只是一些简单的方式，用来展示一个研究思路是如何产生的。一旦你掌握了其中某几个范式，想出新的研究思路将是小菜一碟（当然，在决定实现这个想法之前，仍然需要对它的价值进行评估）。

范式1：填补空白

这是我大学期间，在微软的Harry Shum博士关于研究的报告上听到的一个简单的范式。这个想法在概念上其实很简单。读几篇关于某一主题的论文，记下这些论文在系统、方法、技术、数据集等方面提供的前提假设、证明支撑/属性特点等方面的差异，即下面所提到的“维度”，然后画一张表（不一定是二维的）来进行“维度”比较。找到其中的“空白”，即未被考虑或涉足的部分，这些都是潜在的新的研究思路（事实上，许多好的论文使用这样一个表格来清晰地表明自身的研究与相关研究的区别）。

以对抗机器学习领域为例，许多研究人员最初关注的是图像。例如，如何扰乱一张猫的照片，使它被错误地识别为香蕉。很快，其他领域诸如视频、音频和恶意软件也被拿来做对抗研究（例如，如何通过注入无意义的指令来欺骗恶意软件分类器）。另一个简单的例子是：研究人员可能已经应用静态分析技术来自动发现某些类型的漏洞，但还没有人应用动态分析技术来做这件事。接下来，你可以调研静态分析与动态分析在发现特定类型漏洞上的优缺点。当然，现实中也存在各种类型的静态分析和动态分析技术。你可以制作一个更为细粒度的表格来发现更多的“空白”部分。同样地，如果你已经对某种特定类型的漏洞进行过研究，还可以对其他不同类型的漏洞进行研究。

表 1 填补空白范式示例

成功应用此范式的关键是绘制出该对比空间中存在的“维度”。你研究得越深，就越有可能找到可以被研究的“空白”部分。例如，你必须熟悉不同种类的程序分析技术，然后才能绘制出一张表格并确

定某个特定的技术并没有被用来解决过某个问题。同样地，你需要了解不同类型的漏洞（仅内存损坏漏洞就有十几种类型）。

识别“维度”通常有两种常用的方法。第一种方法，广泛阅读大量论文，寻找类似主题论文之间的差异。相信我，当有一天需要用到相关的论文知识时，这将会很方便。另一种方法，阅读综述文章，因为它们通常已经在多个“维度”上进行了总结梳理。在网络安全领域，IEEE安全与隐私会议（四大安全顶级会议之一）每年以知识体系化（SoK）的形式接收并发表若干篇论文。如果你感兴趣的话，它们绝对值得一读。

范式2：扩充延伸

这是“填补空白”范式的自然递进。正如上文所提到的，绘制空间中一些好的“维度”可能是最具挑战性的一步。但是，如果你在某个方向已经有了一些研究思路（例如发表了一两篇论文），就可以从特别的角度看到别人不容易找到的“维度”。

我们发表在USENIX Security 2016上进行TCP侧信道攻击的论文“Off-Path TCP Exploits: Global Rate Limit Considered Dangerous”（路径外TCP漏洞：全局速率限制是危险的）就是基于这种范式驱动的。我们之前的工作（发表于S&P 2012和CCS 2012），对攻击条件要求非常高（有人可能认为这是不现实的），即假设一个非特权级别的恶意软件已经安装在受害者的智能手机上（或者网络中部署了某种类型的防火墙）。而这篇论文则努力减轻对这样要求苛刻条件的依赖，这也使得我们发现了一个全新的攻击侧信道。实际上，我已经基本绘制出了关于攻击需求的“维度”，即“攻击需求：恶意软件｜防火墙｜无”。

我们发表在CCS 2020的论文“DNS Cache Poisoning Attack Reloaded: Revolutions with Side Channels”（重新加载 DNS 缓存中毒攻击：侧信道革命）也属于这一类。这是另一篇侧信道文章，在文中我们将侧信道经验从TCP迁移到UDP，漏洞的利用本质其实是非常相似的。实际上，这两篇文章是在Linux内核实现中寻找TCP/UDP套接字之间的共享资源（即全局变量）。所以这里的“维度”是关于不同类型的网络协议。

范式3：造锤找钉

该范式也是在Harry的演讲中学到的，我将其归纳进行介绍。概括性的想法是，如果你拥有独一无二的专业知识、技术、系统、甚至数据集（其他人无法轻易得到的），你就可以充分利用它们寻找有趣的问题来进行解决（幸运的是计算机科学领域中有许多这样的实际问题）。例如，密歇根大学Peter Chen教授的团队在虚拟机方向（以及其他方向）拥有充足的专业知识，并且开发了许多有趣的应用程序。如果我没记错的话，Peter的团队构建了世界上第一个完整的虚拟机记录和重播功能。这对于调试、入侵追踪具有十分重要的作用，该领域已经发表许多高质量的论文（如OSDI、ASPLOS、PLDI)。

我的团队也在网络侧信道方向发表了一系列论文，从起始于2012年的TCP到2020年的UDP和DNS方向（在S&P、USENIX Security和CCS发表了7篇论文）。在这个过程中我们掌握了关于发现网络侧信道充足的专业知识。网络侧信道是一个公认的小领域，没有太多的竞争。通过这种方式积累专业知识的好处是，一旦你深入到某一个主题中，你就会更容易找到新的问题来解决。

在网络安全领域中，UCSB大学研究人员开发的名为Angr的系统也值得一提。Angr是现今技术最为成熟的二进制分析框架（包括一个符号执行引擎），最初是为DARPA计算机网络挑战大赛（一个百分百自动化CTF竞赛）开发的。自2016年论文发表以来，该系统一直是开源的。由于它的工程设计和开发都十分完善，很快就在学术界（截至2020年12月28日已被引用540次）和工业界中流行起来。事实上，我们也在一些项目中使用了它。作者自己也利用该工具为多种安全应用构建了后续项目。

最后一个例子是关于数据集的，即UCSD大学的应用互联网数据分析中心(CAIDA)。CAIDA的研究人员开发并部署了若干个基础测量设施，持续运行着多个互联网测量项目，收集大量的互联网数据。由于其他研究人员并未拥有这些数据（至少在学术界），他们能够基于这些大量的独特的数据集展开很多有趣的测量研究。

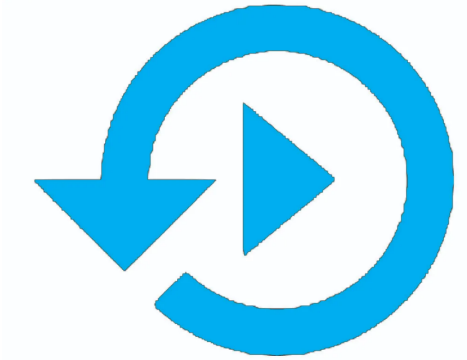
在我看来，尽管这是一种很好的研究方式，有一定的影响力并且可持续研究，但并不一定适用于所有人。首先,构建专门知识、系统或基础设施直到开始有收益,可能会非常耗时。其次，这类计划有时需要整个团队来进行构想、计划和组织（通常超过博士生个人的能力）。最后，只有少数群体主导着一个领域，除非你有独特的视角，否则要超越他们是非常困难的。换句话说，如果你能找到很多人需要的东西，但目前还没有好的解决方案，那么它可能是值得考虑的。

范式4：以小见大

一个研究想法通常从一些微小或偶然的发现开始，而你需要将这个想法深入研究以确定其是否足以支撑一篇可被发表的文章。确认是否需要深入研究或投入多少时间进行深入研究是需要技巧的。根据我的经验，下述信号预示着这个想法或许是值得深入研究的：1.当你第一次发现该现象时（不管这是一个多么细微的现象），你觉得这个现象是如此有趣、如此让你惊喜；2.当你深入挖掘时，你发现这一现象背后有深层次的解释（如某一安全漏洞是由某一类型的设计缺陷导致的）；3.你能发现与之类似的现象。

以我所在团队发表在CCS 2016上的论文“Android ION Hazard: the Curse of Customizable Memory Management System”（Android ION的危害：可定制内存管理系统的诅咒）为例，在测试安卓系统上可能被潜在的恶意程序利用的接口时，我的学生Hang偶然发现了一个有趣的设备文件“/dev/ion”：该接口允许任何应用程序在“预先存在的堆”中分配内存并将其映射至用户空间。令人惊讶的是，我们发现该接口所返回的内存并未被“清零”，即能看到该片内存的使用者之前存储在内存中的数据（信号1）。这意味着我们发现了一个信息泄露的漏洞，尽管该漏洞类型已经存在而且漏洞自身并不能直接形成一个研究项目，但进一步的研究发现严重性远不止于此：该接口的引入暴露了操作系统内核所使用的内存（信号2）。确认这一点是基于如下事实：与被设计用来与用户态软件交互的接口不同，出于性能考虑这类内部接口不会主动将重新分配的内存清零。更糟糕的是，不同的安卓智能手机对“/dev/ion”有其自定义的实现——这也意味着该接口存在着更大的研究空间（信号3）。

范式5：推陈出新



你可能会想：如何才能找到这些细微（却有潜力）的想法？方法之一便是复现前人论文的成果。事实上，论文中的结果和你所复现的结果可能并非完全一致。其背后的原因可能是：

1.作者无意中所犯下的错误；

2.结果本身便不是能够百分之百复现的，如对某些随时间变化的互联网现象的测量；

3.论文中提出方法的基准有失偏颇或数据集相对片面。事实上，前人工作的局限意味着提升的空间。即使你设法按照预期百分之百地复现出了论文中相同的结果，在复现中你也可能得到全新的灵感。

对于上述范式，在由清华大学牵头、我所在的团队深度参与的发表在USENIX Security 2020的论文“Poison Over Troubled Forwarders： A Cache Poisoning Attack Targeting DNS Forwarding Devices”（对问题转发器投毒：针对DNS转发器的缓存投毒攻击）的相关研究中，学生曾被要求复现 CCS 2018的论文“Domain Validation++ For MitM-Resilient PKI”（针对MitM-Resilient PKI的域验证++）。CCS 2018 的文章使用了一种特殊的测量手段来展示其所提到的攻击方法的普适性。有趣的是，复现的结果要比原文所展示的结果消极许多。对造成这一消极结果的原因的深入挖掘促成了一种可以绕过现有防范体系的新的攻击方法的提出。

范式6:外部资源——从工业界与新闻信息中吸取营养

对于网络安全这种实践性较强的领域，与业内人士建立联系、了解其需求和痛点可能会给你的研究带来全新思路。相比学术界，工业界拥有更多的资源，但其思维方式与处理技术问题的优先级却和学术界有所不同：工业界通常强调抗拒风险，对方案的可靠性要求较高。而学术界的好处是：探索性的工作是被允许的，不会要求一次性地解决所有的问题。如果解决某个问题，某种意义上甚至可以自己定义“解决”的标准。如果你正在研究一个行业中亟待解决的问题，并且对该问题尚无完善的解决方案，那么对这个研究项目来说，其“成功”门槛（相对其他较为成熟的领域）会大为降低。
我个人经常会从外部资源得到的灵感中获益。几年前，通过与业内人士的交谈，我发现软件的“打补丁”流程中存在着巨大的问题。以Linux和安卓内核为例，当一个补丁被提交到上游Linux内核中时，下游内核分支（如Linux LTS、Ubuntu和Android）的维护者需要手动检查这些补丁是否需要“应用”到他们的内核。这是一个耗时且容易出错的过程，重要的安全补丁可能会延迟应用甚至直接丢失。更糟糕的是，下游内核分支的使用者很难对内核分支的补丁情况进行审核。以安卓为例，绝大多数内核分支供应商并不提供其源代码的完整提交记录。这促使了自动化测试二进制内核中是否存在补丁的工具的开发，该项工作被发表在USENIX Security 2018上“Precise and Accurate Patch Presence Test for Binaries”（精确的二进制补丁存在性测试）。随后，我们完善了该工具，并在USENIX Security 2021上发表了另一篇论文“An Investigation of the Android Kernel Patch Ecosystem”（Android内核补丁生态系统调研），进一步研究了补丁传播缓慢的根本原因。

而新闻作为另外一个外部资源，同样能够给人带来灵感。事实上，我从几位教授那里学到了这一点，他们使用推特作为其科技新闻源。我们做的一个有趣的项目就是这样从“民情”中得到了启发，这项工作发表在CCS 2015 “Android Root and its Providers: A Double-Edged Sword”（Android Root及其提供商：一把双刃剑）。当时“root”安卓手机很流行，用户借此自定义操作系统并解锁非“root”时无法实现的新功能。在这样的需求下，许多“一键root”应用程序被开发出来，而这些应用程序所适配的手机型号种类繁多（root本质是获取系统最高权限的过程）。“一键root”本质上是对手机操作系统内核所发起的（便利的）攻击。我不禁思考，“一键root”应用程序的开发者究竟掌握了多少能够提升权限的漏洞？其中是否存在未被公开的漏洞？攻击者是否能够窃取这些漏洞并发动攻击（比如勒索软件）？经研究发现，这些root应用程序中有一部分是由业内顶级黑客开发的，其可利用的漏洞超过100种，而攻击者确实能够通过某些手段（如逆向工程）窃取并利用这些漏洞，这是非常可怕的。

网络安全研究中特有的其他范式：

> 对抗性研究。由于安全在本质上是攻防对抗，你总是可以尝试攻破现有的防御机制或者针对已有的攻击技术建立防御方案。实际上，我经常看到一篇新奇的攻击论文发表后，相关的防御论文尾随其后（论文可能来自同一团队，也可能来自不同团队）。
>
> 流程自动化。许多系统安全分析（例如，逆向工程、漏洞发现、错误分类和检查补丁是否应用）总是需要一些手工操作，至少在某些设置中是这样。使这些过程（即使是部分）自动化起来的应用技术，如程序分析，具有重要的研究价值。这可能是一种在网络安全之外但在系统安全中非常常见的范式。

还有一些其他的范式没有提到，我鼓励你去思考一篇论文的思路是如何产生的，一有机会就和作者交谈，并关注你喜欢的研究人员，在他们的论文中寻找研究思路（有时是一系列相关的论文）。我相信你会在某一时刻走向你自己的研究之路！

养成思考研究思路的好习惯

到目前为止听起来还不错吧？问题是如果你不去实践，那就只是一句空话。如果没有一个好的习惯，很可能会忘记去实践。这里有几个建议可以帮助你养成这个好习惯：

要认识到一个项目的落地和一个想法的产生及形成是根本不同的。不要完全沉浸于你所进行的项目中并把自己和外界割裂开来（很多学生都会犯这个错误）。要坚持定期阅读论文，尤其是当某一会议刚刚举行、论文大量公布的时候。你可以至少把这些论文的标题都“扫”一遍，并阅读你所感兴趣的论文的摘要。如果阅读摘要后你发现这篇论文确实是你感兴趣的，那你应该将其更为深入地阅读。你需要确保自己了解论文作者的研究思路，而不能仅仅停留在了解论文中所提到的技术的层面。

你需要严肃对待论文审稿。导师经常将论文审稿的工作交给学生，然后与学生讨论这些待审稿的论文。参与论文审稿是学习“如何写好一篇论文”的好机会。这是因为在通常情况下，你只会阅读到新奇、有影响力的，且文风优雅的论文；而论文审稿时，你将通过审视被拒收的论文，了解其为何被拒收，进而提升你的论文写作能力。

你需要对不同领域的论文保持好奇和开放的心态。我的建议是，你应该“博览群书”。计算机科学的每个单独的领域（如系统、网络、安全、软件工程）都处在越来越成熟的阶段，许多令人惊艳的想法来自于不同领域交叉时碰撞出的思维火花。在我读书期间，尽管我的导师的主要研究方向是计算机网络，但我一直在研读系统和程序分析方向的论文，我认为这种所谓“跨领域”的阅读对我后来的教授生涯帮助很大。

你需要多参加阅读小组、多参与讨论并多提出问题。对于一些讨论量很大的会议（如你所在的研究小组所举办的会议），一定要尝试参加。如果你害怕表达自己的意见（这可能是因为你觉得自己知识不足或经验欠缺），你需要记住：你的导师正在努力帮助小组中的每一个人（尤其是低年级学生）成长和进步，没有人会因为你问一个所谓“愚蠢”的问题而嘲笑你。如果你真的觉得腹中无物不知从何问起，那在讨论前你也可以试着阅读将要讨论的论文。这样，你至少可以在会上讨论下“何种思维令该文章得以成型”。当你经过这样的一两次尝试后，你会感受到这样做的回报（即获得了能进行良好讨论的能力）。长此以往，你便会越来越适应“讨论”这件事本身。对我个人而言，我非常喜欢的事情之一就是辩论某个想法是否“足够优秀”（尽管其已经在相当好的会议上发表）。一些学生扮演攻击者的角色，指出这个想法的弱点和局限性；其他学生则负责“捍卫”这个想法。这种论文审稿模拟有助于让你养成判断某个想法是否足够完善的眼力。

多和你的实验室伙伴们聊天，去了解他们并和他们保持良好的关系。你和他们共处的时间可能会超过你和你的导师相处的时间，所以为什么不让这种共处成为一种愉悦的体验？就某一个研究课题开展一场天马行空的对话，就某篇论文进行一个小小的辩论，在交流中碰撞出思维的火花——这种感觉真是太棒了。建立起这种人际关系的方法之一是定期与他们交谈：向他们询问项目进展，提供给他们真诚的反馈。他们很可能会用相同的方式回馈于你，最终有利你的研究进展。除此之外，当你因为某些原因而感到“卡住”时，和他人的交谈可能能带给你一个审视问题的全新视角，进而帮助你走出困境——我自己便曾在这样的场景中获益。

【作者简介】
钱志云是加州大学河滨分校 (University of California, Riverside) 的副教授。他的研究兴趣在于网络，操作系统，以及软件安全。其中涉及到TCP/IP协议的设计与实现，安卓操作系统的漏洞挖掘和分析，以及测信道在网络系统领域中的安全性研究。研究曾获得 ACM CCS 2020 Distinguished Paper Award, IRTF Applied Networking Research Prize, Facebook Internet Defense Prize Finalist, 以及 GeekPwn 最大脑洞奖。

英文原文链接：
https://zhuanlan.zhihu.com/p/341685279

更多详情请参见https://www.inforsec.org/wp/?page_id=1211，或扫描下方二维码访问InForSec学术视频专区，观看作者该报告视频

（作者为加州大学河滨分校副教授）

来源：中国教育网络

致 谢:

百度安全 有AI更安全

奇安信 新一代网络安全领军者

蚂蚁集团 ANT GROUP

阿里安全 ALIBABA SECURITY

文章已于2021/08/16修改

喜欢此内容的人还喜欢

闲来无事，反制GOBY

赛博回忆录

暴露会话Cookie的CNAME伪装机制

安全学术圈

腾讯安全科恩实验室推出首款免费在线SCA平台：BinaryAI

腾讯科恩实验室