usenix
THE ADVANCED
COMPUTING SYSTEMS
ASSOCIATION

Investigating Commercial Pay-Per-Install and the Distribution of Unwanted Software

Kurt Thomas, Juan A. Elices Crespo, Ryan Rasti, Jean-Michel Picod, Cait Phillips, Marc-André Decoste, Chris Sharp, Fabio Tirelo, Ali Tofigh, Marc-Antoine Courteau, Lucas Ballard, Robert Shield, Nav Jagpal, Moheeb Abu Rajab, Panayiotis Mavrommatis, Niels Provos, and Elie Bursztein, *Google;* Damon McCoy, *New York University and International Computer Science Institute*

https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/thomas

This paper is included in the Proceedings of the
25th USENIX Security Symposium

August 10–12, 2016 • Austin, TX

ISBN 978-1-931971-32-4

Investigating Commercial Pay-Per-Install and the Distribution of Unwanted Software

Kurt Thomas[◇] Juan A. Elices Crespo[◇] Ryan Rasti[◇] Jean-Michel Picod[◇] Cait Phillips[◇]
Marc-André Decoste[◇] Chris Sharp[◇] Fabio Tirelo[◇] Ali Tofigh[◇] Marc-Antoine Courteau[◇]
Lucas Ballard[◇] Robert Shield[◇] Nav Jagpal[◇] Moheeb Abu Rajab[◇] Panayiotis Mavrommatis[◇]
Niels Provos[◇] Elie Bursztein[◇] Damon McCoy[†*]

[◇]*Google* [†]*New York University* [*]*International Computer Science Institute*

Abstract

In this work, we explore the ecosystem of *commercial pay-per-install* (PPI) and the role it plays in the proliferation of unwanted software. Commercial PPI enables companies to bundle their applications with more popular software in return for a fee, effectively commoditizing access to user devices. We develop an analysis pipeline to track the business relationships underpinning four of the largest commercial PPI networks and classify the software families bundled. In turn, we measure their impact on end users and enumerate the distribution techniques involved. We find that unwanted ad injectors, browser settings hijackers, and "cleanup" utilities dominate the software families buying installs. Developers of these families pay $0.10–$1.50 per install—upfront costs that they recuperate by monetizing users without their consent or by charging exorbitant subscription fees. Based on Google Safe Browsing telemetry, we estimate that PPI networks drive over 60 million download attempts every week—nearly three times that of malware. While anti-virus and browsers have rolled out defenses to protect users from unwanted software, we find evidence that PPI networks actively interfere with or evade detection. Our results illustrate the deceptive practices of some commercial PPI operators that persist today.

1 Introduction

In recent years, *unwanted software* has risen to the forefront of threats facing users. Prominent strains include ad injectors that laden a victim's browser with advertisements, browser settings hijackers that sell search traffic, and user trackers that silently monitor a victim's browsing behavior. Estimates of the incident rate of unwanted software installs on desktop systems are just emerging: prior studies suggest that ad injection affects as many as 5% of browsers [34] and that deceptive extensions escaping detection in the Chrome Web Store affect over 50 million users [17].

Despite the proliferation of unwanted software, the root source of installs remains unclear. One potential explanation is *commercial pay-per-install* (PPI), a monetization scheme where software developers bundle several third-party applications as part of their installation process in return for a payout. We differentiate this from *blackmarket pay-per-install* [4] as commercial PPI relies on a user consent dialogue to operate aboveboard. Download portals are a canonical example, where carelessly installing any of the top applications may leave a system bloated with search toolbars, anti-virus free trials, and registry cleaners [16]. Unfortunately, this all too common user experience is the profit vehicle for a collection of private and publicly companies that commoditize software bundling [15]. While earnings in this space are nebulous, one of the largest commercial PPI outfits reported $460 million in revenue in 2014 [31].

In this work, we explore the ecosystem of commercial PPI and the role it plays in distributing the most notorious unwanted software families. The businesses profiting from PPI operate *affiliate networks* to streamline buying and selling installs. We identify a total of 15 PPI affiliate networks headquartered in Israel, Russia, and the United States. We select four of the largest to investigate, monitoring each over a year long period from January 8, 2015–January 7, 2016 in order to track the software families paying for installs, their impact on end users, and the deceptive distribution practices involved.

We find that commercial PPI distributes roughly 160 software families each week, 59% of which at least one anti-virus engine on VirusTotal [36] flags as unwanted. For our study, we use this labeling to classify unwanted software. The families with the longest PPI distribution campaigns include ad injectors, like Crossrider, and scareware that dupes victims into paying a subscription fee for resolving "dangerous" registry settings, a hair's length shy of ransomware. We find that PPI networks support unwanted software as first-class partners: downloaders will actively fingerprint a victim's machine in order to detect hostile anti-virus or virtualized environments, in turn dynamically selecting offers that go undetected. Software developers pay between $0.10–1.50 per install for these services, where price is dictated by geographic demand.

Via Safe Browsing telemetry, we measure the impact of commercial pay-per-install on end users across the globe. On an average week, Safe Browsing generates over 60 million warnings related to unwanted software delivered via PPI—three times that of malware. Despite these protections, estimates of unwanted software incident rates provided by the Chrome Cleanup Tool [5] indicate there are tens of millions of installs on user systems. Of the top 15 families installed, we find 14 distribute via commercial PPI.

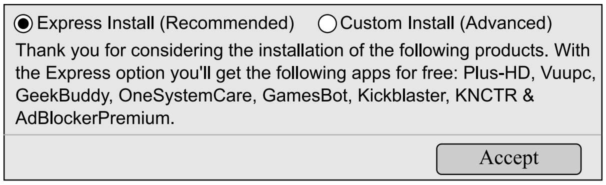
Thousands of PPI affiliates drive these weekly downloads through a battery of distribution practices. We find 54% of sites that link to PPI bundles host content related to freeware, videos, or software cracks. For the long tail of other sites where users are not expecting an installer, PPI networks provide affiliates with "promotional tools" such as butter bars that warn a user their Flash player is out of date, in turn delivering a PPI bundle. In order to avoid detection by Safe Browsing, affiliates churn through domains every 7 hours or actively cloak against Safe Browsing scans. Our findings illustrate the deceptive behaviors present in the commercial PPI ecosystem and the virulent impact it has on end users.

In summary, we frame our contributions as follows:

- We present the first investigation of commercial PPI's internal operations and its relation to unwanted software.
- We estimate that commercial PPI drives over 60 million download attempts every week.
- We find that 14 of the top 15 unwanted software families distribute via commercial PPI.
- We show that commercial PPI installers and distributors knowingly attempt to evade user protections.

2 Commercial Pay-Per-Install

For the purposes of this study, we define *commercial pay-per-install* (PPI) as the practice of software developers bundling several third-party applications in return for a fee. We present an example bundle in Figure 1, where clicking on "accept" results in a user installing eight *offers* through a single radio dialogue. Some of these offers may be *unwanted software*, where at least one anti-virus engine on VirusTotal marks the application as potentially unwanted, adware, spyware, or a generic category. In contrast to blackmarket pay-per-install which illegally sells access to compromised hosts, deceptive commercial PPI outfits rely on this prompt to nominally satisfy user consent requirements. To simplify the process of buying and selling installs, commercial PPI operates as an affiliate network. We outline this structure and enumerate the major networks in operation during our study.



Thank you for considering the installation of the following products. With the Express option you'll get the following apps for free: Plus-HD, Vuupc, GeekBuddy, OneSystemCare, GamesBot, Kickblaster, KNCTR & AdBlockerPremium.

Figure 1: Sample prompt bundling eight commercial pay-per-install offers. Each offer is downloaded and automatically installed upon a user accepting the "Express Install" option. Users may have no knowledge of the behaviors of the bundled offers.

2.1 PPI Affiliate Structure

The pay-per-install affiliate structure consists of *advertisers*, *publishers*, and *PPI affiliate networks*. Figure 2 presents the typical business role each plays.

Advertiser: In the pay-per-install lingo, advertisers are software owners that pay third-parties to distribute their binaries or extensions. Restrictions on what software advertisers can distribute falls entirely to the discretion of PPI affiliate network operators and their ability (and willingness) to police abuse. As highlighted in Figure 2, advertisers include developers of unwanted software like Conduit, Wajam, or Shopperz that recuperate PPI installation fees by monetizing end users via ad injection, browser settings hijacking, or user tracking. Irrespective of the application's behavior, PPI networks set a minimum bid price per install that advertisers only pay out upon a successful install. Advertisers may also restrict the geographic regions they bid on.

Publisher: Publishers (e.g., affiliates) are the creators or distributors of popular software applications (irrespective of copyright ownership). An example would be a website hosting VLC player as shown in Figure 2. PPI networks re-wrap a publisher's application in a *downloader* that installs the original binary in addition to multiple advertiser binaries. This separation of monetization from distribution allows publishers to focus solely on garnering an audience and driving installs through any means. Consequently, advertisers may have no knowledge of the deceptive techniques that publishers employ to obtain installs, nor what their binary is installed alongside. Upon a successful install, the publisher receives a fraction of the advertiser's bid. We differentiate this from direct distribution licenses such as Java's agreement to bundle the Ask Toolbar [18], as there is no ambiguity be-

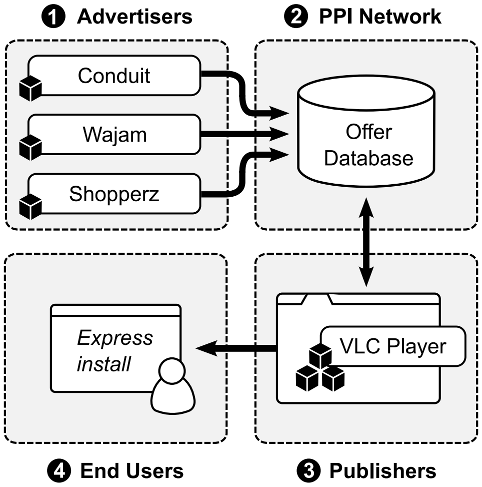


Figure 2: Pay-Per-Install (PPI) business model. Advertisers paying for installs supply their binaries to a PPI affiliate network (❶). The PPI network cultivates a set of publishers—affiliates with popular software applications seeking additional monetization (❷). The PPI network re-wraps the publisher's software with a customized downloader that the publisher then distributes (❸). When end users launch this downloader, it installs the publisher's software alongside multiple advertiser binaries (❹). The PPI network is paid by the advertisers and the publisher receives a commission.

tween the advertiser and publisher around what packages are co-bundled and the source of installs.

PPI Affiliate Network: PPI affiliate networks serve as a bridge between the specialized roles of advertisers and publishers. The PPI network manages all business relationships with advertisers, provides publishers with custom downloaders, and handles all payments to publishers for successful installs. When a publisher gains access to an end user's system, the PPI network determines which offers to install. As we show in Section 3, this entails fingerprinting an end user's system to determine any risk associated with anti-virus as well as to support geo-targeted installations. Similarly, the PPI network dictates the level of user consent when it installs an advertiser's binary, where consent forms a spectrum between silent installs to opt-out dialogues. In some cases, advertisers can customize the installation dialogue and thus play a role in user consent.

Reselling: With multiple PPI affiliate networks in operation, various PPI operators will aggregate their publishers' install traffic and resell it to larger PPI affiliate networks. These smaller PPI operators create value for their affiliates by providing promotional tools in the form of landing pages, banner ads, butter bars (e.g., "Your Flash player is out of date"), and generic installers for media players and games—described later in Section 6. These tools simplify the process of monetizing web traffic where a victim is not primed to download a bundle. It is worth noting that these resellers do not operate their own downloader; they rely on sub-affiliate tracking provided by larger PPI networks that effectively enables two-tiered affiliate distribution.

PPI Affiliate Network	First Seen	Reseller
AirInstaller	09/2011	
Amonetize	01/2012	
InstallCore	04/2011	
InstallMonetizer	06/2010	
InstallMonster	06/2013	
Installaxy	06/2014	✓
Installerex	12/2013	✓
NetCashRevenue	01/2014	✓
OpenCandy	04/2008	
Outbrowse	11/2012	
PerInstallBucks	06/2013	✓
PerInstallCash	04/2011	✓
Purebits	06/2013	✓
Solimba	08/2013	
Somoto	10/2010	

Table 1: List of 15 PPI affiliate networks, an estimate of when they first started operating, and whether they resell installs.

2.2 Identifying PPI Networks

In contrast to blackmarket pay-per-install [4], the affiliate networks driving commercial PPI are largely private companies with venture capital backing such as *InstallMonetizer* and *OpenCandy* [8, 9]. Registering as a publisher with these PPI networks is simple: a prospective affiliate submits her name, website, and an estimate of the number of daily installs she can deliver. Given this porous registration process, underground forums contain extensive discussions on dubious distribution techniques and which PPI affiliate networks offer the best conversion rates and payouts. We tracked these conversations on *blackhatworld.com* and *pay-per-install.com*, enumerating over 50 commercial PPI affiliate programs that exclusively deal with Windows installs. While there are networks that target Mac and mobile installs, we focus our work on the relationship between commercial PPI and unwanted software families that disproportionately impact Windows users as identified by previous studies [17, 34].

2.3 Acquiring PPI Downloader Samples

As part of our initial investigation of PPI, we successfully acquired downloaders for fifteen distinct PPI networks. We list each in Table 1. These networks have been in operation for an average of 2–3 years, with the oldest program dating back to 2008 as gleaned from crawl logs provided by *archive.org*. Based on a preliminary black-

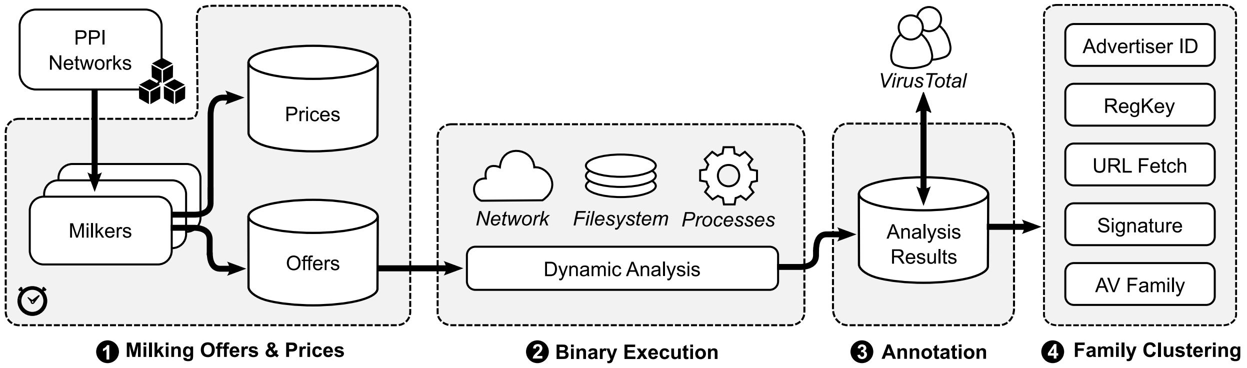


Figure 3: PPI monitoring infrastructure. We collect offers and prices from four PPI networks on a regular basis (❶). We then execute the offer binaries in a sandbox to observe network requests, file system changes, and running processes (❷). We annotate each binary with any known VirusTotal labels (❸) before finally clustering binaries into families (❹).

box test of each downloader, we found six of the fifteen PPI downloaders were merely resellers for other PPI networks in our list. Of the remaining nine, we elect four of the largest—Amonetize, InstallMonetizer, OpenCandy, and Outbrowse—as the basis of our investigation into the role of PPI in unwanted software distribution. We based our initial selection criteria on the complexity of the offer protocols and from preliminary statistics reported by Safe Browsing on which PPI networks delivered the largest number of downloads. We confirm later in Section 5 that these four PPI networks are in fact representative, large operators. We also explore the impact of the PPI ecosystem as a whole on end users.

3 Monitoring the PPI Ecosystem

Using the PPI downloader samples we acquire for Amonetize, InstallMonetizer, OpenCandy, and Outbrowse, we develop a pipeline to track the offers (e.g., advertiser binaries) that each PPI network distributes as well as the regional price per install. We outline our pipeline in Figure 3. We begin by simulating each PPI downloader's protocol to fetch all possible offers on an hourly basis. We analyze each binary in a sandboxed environment, ultimately clustering the offers into software families based on the behavioral patterns we observe. We discuss the construction of our pipeline and its limitations.

3.1 PPI Downloader Protocol

All four PPI downloaders we study rely on a three-stage protocol for dynamically fetching advertiser binaries. To start, a downloader fingerprints a client's device to determine the operating system and default browser. The downloader reports these parameters to the PPI server as part of a request for all available offers as shown in Figure 4. In our example, the request embeds the exact version of the client's OS and service pack; the Chrome, Firefox, and Internet Explorer version if any are present; whether the system is 32-bit or 64-bit; and finally potentially unique identifiers including a MAC address and a machine identifier such as `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography`.

We provide a typical offer response in Figure 5. Each offer contains a unique product identifier, download URL, and additional metadata that dictates how the install process unfolds. Depending on the PPI network, the response will include anywhere from 5–50 offers, filtered by regional requirements imposed by the PPI server based on the client's IP address.

In the second stage, the downloader verifies that none of the `RegKey` or `AntivirusesRegKeys` are present in the device's registry. This approach serves to prevent multiple installations of the same advertiser's binary as well as to avoid anti-virus disrupting the installation of an offer. If a client's machine satisfies the offer criteria, the downloader will display the offer and execute the binary with the command line options specified by the PPI server if accepted. These parameters sometimes reveal the intent of the advertiser (e.g., replacing the default search provider) as well as evasive actions such as remaining dormant for 20 days to prevent unwanted software symptoms (e.g., injected ads) from manifesting immediately after an installation. If a client's system does not satisfy the criteria, the downloader simply tests another potential offer until all options are exhausted. In total, the downloader will display offers for anywhere from 1–10 advertiser binaries (potentially all as one express install dialogue): the maximum is dictated by the PPI network.

In the final stage, the downloader reports all successfully installed offers along with the publisher's affiliate id for compensation.

```
http://srv.desk-top-app.info/Installer/
Flow?os=6.1&ospv=-1&iev=9.11&ffv=&
chromev=46.0&macaddress=00:00:00...
&systembit=32&machineguid=b1420e...
```

Figure 4: Example PPI network request for Outbrowse containing the components that make up a device fingerprint.

```
{
  "SleepAfterInstall": 1800000,
  "ExeURL": "http://example.com/file7",
  "AntivirusesRegKeys": "[
      {"RegKey32": "...\\McAfee..."}],
  "RegKey": ...,
  "PostRegKey": ...,
  "ProductID": 10001,
  "CommandLine": "-defaultsearch=true",
  "RunInAggressiveInstaller": "1",
}
```

Figure 5: Example PPI network response for Outbrowse containing an offer and associated metadata.

PPI Network	Sample Domain
Outbrowse	`srv.desk-top-app.info`
Amonetize	`www.download-way.com`
InstallMonetizer	`www.stsunsetwest.com`
OpenCandy	`api.opencandy.com`

Table 2: Sample of PPI server domains contacted by our milkers. In total, we identify 31 domains servicing offer requests for the four PPI networks we study.

PPI Network	Milking Period	Offers	Unique
Outbrowse	1/08/15–1/07/16	107,595	584
Amonetize	1/08/15–1/07/16	231,327	356
InstallMonetizer	1/11/15–1/07/16	30,349	137
OpenCandy	1/09/15–1/07/16	77,581	134
Total	1/08/15–1/07/16	446,852	1,211

Table 3: Breakdown of PPI networks, milking periods, and the unique offers appearing in our dataset.

3.2 Developing Longitudinal Milkers

In order to track bundled offers, we develop *milkers* that replay the first stage of each PPI network's offer protocol and decode the response. This is largely a time-intensive manual process of blackbox testing each PPI downloader, determining the PPI server's domain (or those it cycles through), and re-implementing the protocol into a standalone module that generates a network request with the expected downloader User-Agent and custom headers. We provide a sample of the PPI server domains contacted by downloaders in Table 2.

For all requests, we present a device fingerprint associated with a Windows 7 system with Chrome and Internet Explorer installed while randomizing unique identifiers such as the device's MAC address and machine ID. Upon receiving a response, we decode the list of offers and extract the associated URL of the offer binary. We reiterate that the PPI programs we monitor provide anywhere from 5–50 potential offers along with their installation requirements. For each offer, we detect whether we previously observed the URL of the associated binary. If the URL is fresh, we download the URL's content; if the URL is redundant, we rely on a cached copy in order to reduce network load on the PPI servers. We note that this caching methodology may reduce the number of unique digests we obtain if advertisers were to cycle binaries referenced by a fixed URL.

We finally store each binary, the offer metadata (e.g., registry requirements, advertiser ids), and the timestamp of execution. We ran our milkers every hour from a collection of cloud instances hosted in the United States over a year long period from January 8, 2015–January 7, 2016. During this time, we updated our milker protocols at most once per PPI network, a reflection of the lack of external pressure on commercial PPI practices compared to malware. In total, we collected 446,852 offers. These offers contained 2,841 unique URLs, 1,809 unique digests, and 1,211 unique product identifiers (as determined by the `ProductID` field shown in Figure 5, or its equivalent for other PPI networks, which are consistent across versions.) We provide a detailed breakdown of the offers per PPI network in Table 3.

We faced a separate challenge for tracking regional pricing. In particular, the exact daily prices that advertisers pay per install are available only to publishers delivering successful installs. Unlike previous investigations into blackmarket PPI [4], we elected not to register as commercial PPI affiliates due to potential Terms of Service violations. As such, we lack access to per-advertiser pricing data. Instead, we track the average price per install across the PPI ecosystem as publicly advertised by PPI networks and resellers to attract affiliates. In total, we identify five PPI-related websites that provide a breakdown of the current price per install paid across 219 regions, with rates varying between $0.01–$2.09. These sites include `cinstaller.com`, `installmania.com`, `cashmylinks.com`, `perinstallbucks.com`, and `truemediaparnter.com`. We crawled and parsed these pages (as allowed by robots.txt) on a weekly basis from January 8, 2015–January 7, 2016 to monitor any fluctuations.

3.3 Executing and Annotating Offers

We execute all downloaded binaries in a sandboxed environment similar in flavor to Anubis [2], CWSandbox [38], and GQ [22], the details of which are covered in previous work [17, 29, 34]. During execution, we log all network requests and responses, file system changes, modified registry keys, and spawned processes. We also monitor whether the executable changes any preferences related to Chrome or Internet Explorer such as altering the default browser, dropping an extension, or modifying the startup page.

Independent of our dynamic execution environment, we annotate each binary with third-party intelligence gathered through VirusTotal at the end of our collection period. Mechanically, we submit the hash of each binary to determine which of 61 anti-virus engines report the binary as malicious or unwanted. We also collect any labels, though the value of these is highly variable: some reflect generic 'Adware' while others contain a family name potentially unique to an anti-virus engine.

3.4 Clustering and Classifying Offers

At the conclusion of our collection period we classified all of the advertiser binaries in our dataset into distinct families. This canonicalization step is necessary to de-duplicate instances where the same advertiser works with multiple PPI networks or where advertisers introduce polymorphism due to software updates, sub-affiliate programs, or to evade detection by anti-virus engines. Classification is a semi-automated process where we first cluster all binaries based on overlapping registry key modifications, domains contacted during execution, process names, or digital certificates used to sign the advertiser's software (only 58% of offers were signed). This approach follows similar strategies for clustering malware delivered via drive-by downloads [14] and unwanted software using code-signing [21]. We also cluster offers based on the registry keys present in the installation pre-conditions provided by PPI networks during offer selection. These pre-conditions unambiguously reveal all of the registry paths controlled by a single family, such as *Vitruvian* which goes by 19 other names including *LessTabs*, *SearchSnacks*, *Linksicle*; or *Wajam* which installs under 418 unique registry keys. We present a sample of these pre-conditions in Figure 6. Through all these clustering techniques, we generate 873 non-overlapping clusters (of 1,809 possible).

We manually review all clusters active for more than 150 days (e.g., we examine the timestamp of all milked binaries in a cluster and count the number of distinct dates) totaling 58 distinct clusters. We derive family labels based on the most common naming convention found in VirusTotal for a cluster. If no public name exists, we fall back to the advertiser name listed in the offer metadata provided by PPI networks. For all clusters lasting less than 150 days, we rely exclusively on the advertiser name. These names serve only to communicate the major software families commonly found in commercial PPI and whether they overlap with the largest unwanted families impacting end users (discussed in Section 5).

```
this.bCompExist = g.ami.CheckRegKey(
  "Software\\Wajam",
  "Software\\WInternetEnhance",
  "Software\\WajNEnhance",
  "Software\\WWebEnhance",
  "Software\\WaWebEnhance",
  "Software\\WajIntEnhancer",
  "Software\\WajaIntEnhancer",
  "Software\\WNEnhancer",
  "Software\\WajaInternetEnhance",
  "Software\\WInterEnhance",
  "Software\\WajNetworkEnhance",
  "Software\\WajaNetworkEnhance",
  "Software\\WWebEnhancer",
  "Software\\WaWebEnhancer",
  "Software\\WajWebEnhancer",
  "Software\\WajaWebEnhancer",
  .....
  "Software\\Wajam\\affiliate_id")
```

Figure 6: Example offer requirements for Wajam via Amonetize. It contains 418 registry key checks for Wajam variants. We cluster offers that contain the same registry checks.

3.5 Limitations

Our investigation of the PPI ecosystem faces a number of limitations. First, our pipeline runs exclusively from United States IP addresses. This potentially biases our perspective of PPI offers in the event advertisers distribute exclusively to non-US territories. As we demonstrate later in Section 4, the US is the highest paid region for installs, which makes it one of the most interesting to analyze. Next, because we do not participate directly in the PPI ecosystem, we lack exact pricing details per install. We attempt to extrapolate these values based on public pricing used to attract affiliates, but we cannot verify the accuracy of this data other than to corroborate similar rates cited within the underground. Third, our family classification faces the same challenges of malware phylogeny where there is frequent disagreement between anti-virus naming conventions. We reconcile these discrepancies for the longest running PPI campaigns at the expense of overlooking the long tail of brief campaigns. Finally, our perspective of the PPI ecosystem is restricted to four PPI networks due to the time-intensive process of building milkers. While there is a risk our findings are not representative of the entire ecosystem, we show in Section 4 there is substantial overlap between the advertisers of each PPI network. This leads us to believe our sample of PPI networks extends to other unexplored commercial PPI operators.

4 Exploring Commercial PPI Offers

We provide a bird's-eye-view of the business relationships underpinning the commercial PPI ecosystem before diving into the unwanted software families reliant on PPI distribution. We find that ad injectors, browser settings hijackers, and system "clean-up" utilities dominate the advertisers paying for installs. With anti-virus engines flagging 59% of the weekly software families we milk per PPI network, we observe at least 20% of PPI advertisers take advantage of anti-virus and VM detection provided by PPI downloaders to avoid installing in hostile environments.

4.1 High-Level Metrics

Using the 1,211 product identifiers embedded by PPI networks in each offer for accounting purposes, we calculate the total distinct simultaneous offers per PPI network and the duration that advertisers run each offer. On average, we observe 25–60 active offers per PPI network each week, with a fine grained breakdown shown in Figure 7. The spike around July 2015 for Amonetize represents a temporary 2x increase in offers distributed by the PPI network; it is unrelated to any change in our infrastructure. The majority of advertisers for Amonetize and Outbrowse maintain their offers for less than a week before cycling to a new product as shown in Figure 8. In contrast, OpenCandy and InstallMonetizer attract advertisers who run the same product for over 15 days.

4.2 Longest Running Campaigns

With over 873 software families classified by our analysis pipeline, we examine which families consistently appear in the PPI ecosystem and thus sink the most money into installs. Table 4 provides a detailed breakdown of the software families with the longest running distribution campaigns and the PPI networks involved. The families fall into five categories: ad injectors, browser settings hijackers, system utilities, anti-virus, and major brands. We provide sample screenshots of the resulting user experience after installation in the Appendix.

Ad Injectors: Ad injectors modify a user's browsing experience to replace or insert additional advertisements that otherwise would not appear on a website. Every PPI network we monitor participates in the distribution of ad injectors. Of the top eight programs listed by Thomas *et al.* as the largest contributors to ad injection in 2014 for Chrome, Firefox, and Internet Explorer [34], we observe six currently in the PPI ecosystem. The companies behind these software products are commercial entities that span the globe: *Wajam* is located in Canada, *Eorezo* is from France, while *Crossrider* originates from Israel. These ad injectors recuperate the initial sunk cost of installs by monetizing users via display ads and shopping helpers until a victim finally uninstalls the injector.

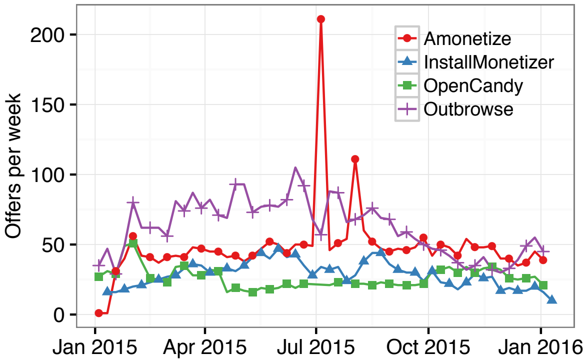


Figure 7: Unique PPI offers operating each week. Amonetize and Outbrowse cultivate a large number of offers compared to OpenCandy and InstallMonetizer.

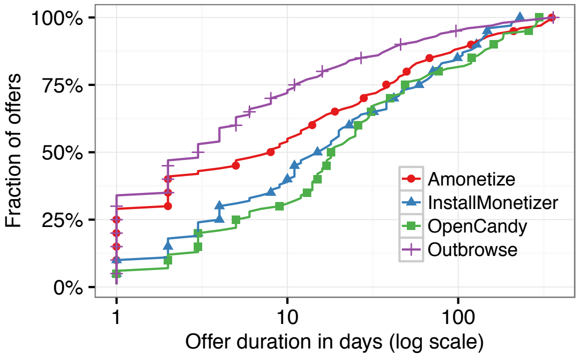


Figure 8: Lifetime of PPI offers. Advertisers run the same offer on OpenCandy and InstallMonetizer for a median of 15 days, while Amonetize and Outbrowse offers quickly churn out of existence to be replaced by new binaries.

Browser Settings Hijackers: Settings hijackers modify a victim's default browser behavior, typically to change the default tab or search engine to a property controlled by the hijacker. These companies subsequently monetize victims by selling their traffic to search engines and potentially tracking user behavior. Examples include Conduit Search (e.g., Search Protect) which came preinstalled on Lenovo machines in 2014 [3]. We note that some hijackers also profit by doubling as ad injectors.

System Utilities: System utilities attempt to upsell users using potentially deceptive practices, with some meeting anti-virus definitions of scareware. This category includes "speedup" utilities like *Speedchecker* and *Uniblue* that present nebulous claims such as "Attention! 2203 items are slowing down your PC" or "your system registry health status is dangerous." These families repeatedly generate pop-up warnings until a victim either pays a subscription fee of $30–40 or uninstalls the software. This scheme is nearly identical to fake anti-virus, but speedup utilities operate under a veil of legitimacy because they remove files from a client's machine, thus satisfying some notion of system improvement. Consequently, anti-virus engines do not consider these families to be malicious, only unwanted. Our categorization also includes cloud backup utilities that repeatedly prompt victims to upload their files to the cloud. Adhering to the dialogue requires victims pay a recurring $120 subscription fee.

All five of the top system utility families are themselves affiliate programs. *Speedchecker* promises affiliates a 30% commission on subscriptions. *Uniblue* advertises a commission of 70%. What emerges is a three-tiered distribution network where system utility affiliates register as advertisers on PPI networks and pay an upfront distribution cost, but reap the commissions on successful subscription conversions. It is also possible that the system utility companies maintain a direct relationship with PPI networks.

Anti-Virus: We observe four anti-virus products distributed via the PPI ecosystem: AVG, LavaSoft, Comodo, and Qihoo. We cannot determine whether these companies directly purchase installs from commercial PPI affiliate networks. We note that all four operate affiliate programs to outsource installs [1, 7, 24, 37]. Assuming all of the installs we observed originate from affiliates, it is unclear how each anti-virus operator polices abuse in the face of an increasingly tangled web of purchased installs and potentially dubious distribution practices. Equally problematic, PPI downloaders simultaneously install these anti-virus products alongside browser settings hijackers and ad injectors—an unenviable user experience.

Major Brands: We observe a small number of major software brands including *Opera*, *Skype*, and browser toolbars distributed via PPI. Based on the affiliate codes embedded in the download URLs for Opera, it appears that Opera directly interacts with PPI operators to purchase installs rather than relying on intermediate affiliates.[1] The other three programs all operate affiliate programs, yielding a similar distribution pattern to that of anti-virus, though we cannot rule out direct relationships with commercial PPI.

Category	Family	Days	Networks	AV
Ad Injector	*Wajam*	365	A, C, I, O	13
Ad Injector	*Vopackage*	365	A, I, O	42
Ad Injector	*Youtube Downloader*	365	A, I, O	50
Ad Injector	*Eorezo*	365	A, O	32
Ad Injector	*Crossrider*	350	A, I, O	55
Ad Injector	*Bubble Dock*	340	O	8
Ad Injector	*Nuvision Remarketer*	322	A	18
Ad Injector	*Download Manager*	313	A	37
Ad Injector	*Vitruvian*	242	A, I, O	41
Hijacking	*Browsefox*	363	A, C, I, O	49
Hijacking	*Conduit*	327	A, I, O	41
Hijacking	*CouponMarvel*	300	A	3
Hijacking	*Smartbar*	294	A, I, O	45
Hijacking	*Safer Browser*	279	A, I, O	3
Utilities	*Speedchecker*	365	A, O	5
Utilities	*Uniblue*	347	A, C, I, O	49
Utilities	*OptimizerPro*	302	A, C, I, O	29
Utilities	*My PC Backup*	292	A, C, I	2
Utilities	*Pro PC Cleaner*	287	A, I, O	33
Utilities	*Systweak*	249	A, I, O	37
Anti-virus	*AVG Toolbar*	333	A, C	0
Anti-virus	*LavaSoft Ad-aware*	305	C	0
Anti-virus	*Comodo GeekBuddy*	153	A, C, I, O	0
Anti-virus	*Qihoo 360*	144	C, I	0
Brand	*Opera*	340	A, C, I, O	0
Brand	*Skype*	176	C, O	0
Brand	*Yahoo Toolbar*	27	O	5
Brand	*Aol Toolbar*	25	O	4

Table 4: Software families with the longest PPI campaigns. We annotate each with the type of software, the days the campaign ran for, the PPI networks involved, and the number of anti-virus engines that flag the family as unwanted. We abbreviate PPI networks as [A]monetize, Open[C]andy, [I]nstallMonetizer, and [O]utbrowse.

4.3 Long Tail of Campaigns

Outside the top 28 longest running PPI campaigns, a question remains on the mixture of credible and unwanted software that makes up the other 845 short lived campaigns. To explore this, we calculate the fraction of software families distributed per week by commercial PPI where at least one anti-virus engine in VirusTotal flags the family as unwanted. Figure 9 presents our results. On an average week, anti-virus engines label 85% of software families distributed by InstallMonetizer

[1] For example, we observe Outbrowse specifically referenced in the target download URL for Opera: `net.geo.opera.com/opera/stable?utm_medium=pb&utm_source=outbrowse&utm_campaign=2328`

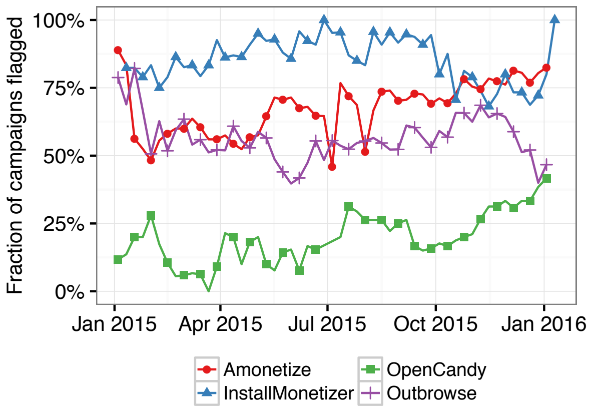


Figure 9: Fraction of software families found each week in PPI networks that were flagged by any anti-virus engine in VirusTotal.

as unwanted, compared to 68% for Amonetize, 57% for Outbrowse, and 20% for OpenCandy. These trends hold true for the entirety of our year-long monitoring. Our findings illustrate that unwanted software dominates both long and short-lived campaigns. The only exception is OpenCandy, which predominantly cultivates advertisers related to games and anti-virus, and to a lesser extent, system utilities and some ad injectors. As a consequence though, OpenCandy has the smallest pool of offers (as discussed previously in Figure 7), while other PPI networks deal with a large number of unwanted software creators and affiliates.

4.4 Contending with Anti-Virus

As discussed in Section 3, each PPI network provides advertisers with a capability to pre-check whether an anti-virus engine is present prior to displaying the advertiser's offer. This pre-check consists of a blacklist of registry keys, file paths, and registry strings specified by the advertiser. We present a sample in Figure 10. To estimate the fraction of offers that take advantage of this capability, we manually collate a list of 58 common anti-virus tokens that appear in a random sample of pre-check requirements, as well as the names of anti-virus companies participating in VirusTotal. We then scanned all offer installation requirements for these tokens.

Of the unique offers in our dataset, 20% take advantage of PPI downloader capabilities that prevent installs from occurring on clients running an anti-virus engine. When anti-virus checks are present, we find advertisers target an average of 3.6 AV families. Our findings suggest that PPI networks support unwanted software developers as first-class partners. We caution our metric is a strict underestimate in the event PPI downloaders scan for side-effects related to anti-virus rather than the exact brand names. We find the most frequently targeted brands include ESET, Avast, AVG, McAfee, Avira, and Symantec. We also observe offers checking for registry keys related to VirtualBox, VMWare, and OpenVPN. There are two possible interpretations of this behavior: advertisers seek to protect themselves from fraudulent installs on virtualized systems; or advertisers actively prevent installations on suspected security testing environments. Given the virtualization checks co-occur with anti-virus evasion, we hypothesize the latter is more likely. Added to our earlier observation that PPI downloaders provide a capability to impose a symptom-free quiet period after installation, a picture emerges of PPI networks actively supporting unwanted software as a first-class partner.

```
g_ami.CheckRegKey(
  "Software\\Avast Software"
  "Software\\Symantec"
  "Software\\KasperskyLAB"
  "Software\\Norton"
  "Software\\Microsoft\\Microsoft Anti.."
  "Software\\Microsoft\\Microsoft Secu.."
  "Software\\Malwarebytes"
  "Software\\Avira")
g_ami.PathExists(
  "%ProgramFiles%\\mcafee"
  "%ProgramFiles%\\Microsoft Security..."
  "%ProgramFiles%\\Malwarebytes...")
```

Figure 10: Example of anti-virus checks performed by a PPI downloader in order to avoid displaying certain offers to clients running hostile anti-virus engines.

4.5 Regional Pricing Per Install

Far and away, installs from the United States fetch the highest price at roughly $1.50 each. The United Kingdom is the second most lucrative region at roughly $0.80 per install. We find that advertisers pay the highest rates for installs from North America, Western Europe, and Japan as shown in Figure 11. Prices outside these regions hover around $0.02–$0.10 per install. This holds true throughout the entirety of our investigation as shown in Figure 12 with relatively little volatility in the market. Despite these lower rates, we show in the next Section that commercial PPI impacts clients around the globe.

5 Measuring User Impact

Through Safe Browsing, we estimate the virulent impact that the PPI ecosystem has on end users. Beginning in 2014, Safe Browsing added support to warn users of Chrome and Firefox against downloading PPI-laden

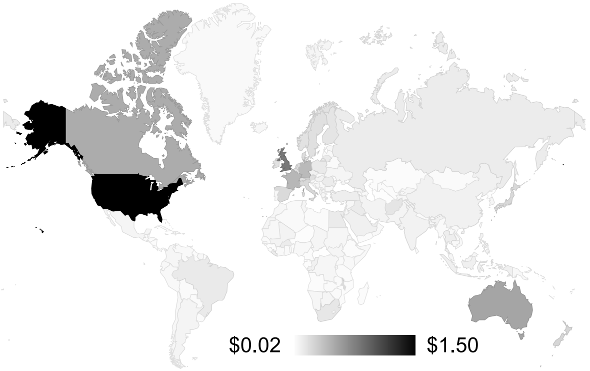


Figure 11: Average price per install across all PPI price monitoring vantage points. Installs from the United States fetch the highest price at $1.50 each.

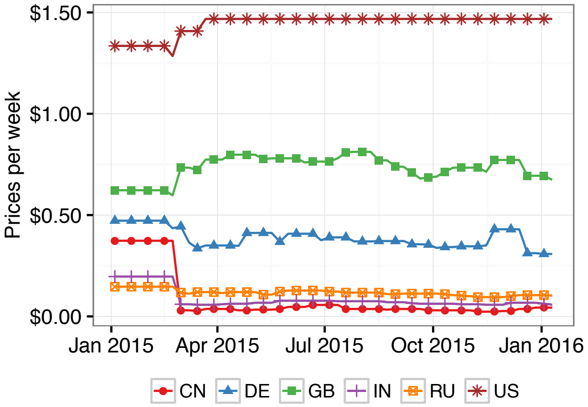


Figure 12: Weekly average price per install for six regions. We observe relatively little volatility for US installs and a slight decline in rates in Europe over time.

software that violates Google's unwanted software policy [28]. This policy covers a subset of applications flagged by anti-virus engines as unwanted. We map these metrics to the PPI networks we study and find that Safe Browsing generates over 60 million weekly download warnings and browser interstitials. Despite these protections, telemetry Chrome users submit about their systems indicate there are tens of million of installations of unwanted software, with nearly all of the top families contemporaneously paying for installs.

5.1 Requests for PPI Downloaders

We rely on two datasets to estimate the volume of weekly downloads to software monetizing through Amonetize, InstallMonetizer, OpenCandy, and Outbrowse: (1) pings reported by browsers integrated with Safe Browsing for downloaded binaries; and (2) Safe Browsing's repository of over 75 million binaries (including benign software). When a browser integrated with Safe Browsing fetches a binary from an untrusted source, it generates an API request to Google in order to obtain a verdict for whether the binary is unwanted or malicious. This request contains hosting details about the binary (e.g., URL, IP address) and related metadata including a digest of the binary [27]. In order to map these downloads to digests of known PPI downloaders, we scan Safe Browsing's repository of dynamic execution traces in search of network requests that match the offer discovery protocol used by each PPI affiliate network (previously discussed in Section 3). From this repository, we identify 1.5 million binaries tied to one of the four PPI networks we study.

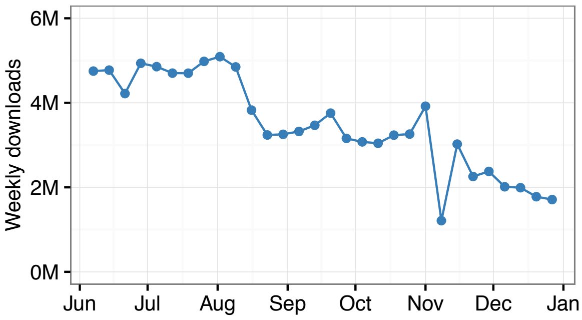


Figure 13: Volume of weekly requests for any of 1.5 million PPI downloaders. We stress this is a lower bound due to missing samples.

We show the total weekly downloads for these 1.5 million binaries between June 1, 2015–January 7, 2016 in Figure 13, irrespective of whether Safe Browsing displayed a warning. We caution these estimates of traffic to PPI networks should serve only as a lower bound as Safe Browsing's coverage of all possible binaries is incomplete. Similarly, due to Safe Browsing displaying warnings for policy-violating PPI downloaders, operators have an incentive to quickly cycle binaries and hosting pages. Caveats aside, we find publishers for the four PPI networks drive an average of 3.5 million downloads per week, though the volume appears to be in decline. Even as a lower bound, our results illustrate the massive influence that PPI networks have on unwanted software distribution.

5.2 PPI Downloader Warnings

In order to obtain a broader perspective of the entire PPI ecosystem's impact on end users (not just the four networks we study), we measure the volume of weekly warnings generated by Safe Browsing for PPI downloaders. Users encounter warnings in one of two ways: download warnings that trigger for policy-violating PPI downloaders, and full-page interstitials that appear when users visit websites commonly distributing PPI-laden software. Because affiliate publishers attempt to evade detection (discussed more in Section 6), Safe Browsing relies on a reputation system called CAMP that builds on incomplete data [29]. The system starts from a seed set of 3 million PPI downloaders that includes samples for all fifteen PPI networks we outlined previously in Section 2. From there, the system scores websites hosting these binaries, common redirect paths, and related binaries. This expands the coverage of sites and binaries involved in pay-per-install, but results in a loss of attribution to individual PPI families. As such, we can only provide an aggregate impact estimate.

We present the volume of PPI downloader warnings and page-level interstitials generated by Safe Browsing between June 1, 2015–January 7, 2016 in Figure 14 and Figure 15 respectively. On an average week, Safe Browsing raises 35 million download warnings and displays 28 million interstitials. Warnings appear as a bursty process, in part due to the arrival of new distribution campaigns and in part due to the ongoing evolutions in the reputation of websites and binaries. In order to place unwanted software in the greater context of threats facing users, we compare the volume of users encountering PPI downloaders versus malware. On average, Safe Browsing raises 13.5 million download warnings and 9 million interstitials to protect users from malware—three times less than that of unwanted software.

The risk of unwanted software is not localized to any single region. We provide a breakdown of the geolocation of users shown a warning related to PPI downloaders in Table 5. We find that Indian users account for 8% of warnings, followed in popularity by Brazil, Vietnam, and the United States. We find no correlation between the price per install and geographic regions with high incident rates. As such, it appears that PPI networks drive installs to any possible user, even when the payout hovers around $0.10 per install.

5.3 Existing Unwanted Installs

For those PPI downloaders that escape detection and launch on a client's machine, we estimate the number of users potentially affected. To do this, we tap into metrics kept by the Chrome Cleanup Tool, an opt-in tool that scans a user's machine for symptoms induced by popular ad injectors, browsing settings hijackers, and system utilities and removes offending programs [5]. Given hundreds of potential unwanted software strains, the tool prioritizes families based on telemetry built into Chrome and system traces supplied by users who file Chrome complaints due to undesirable user experiences. As part of its execution, the tool reports which unwanted software families it identifies as well as those successfully removed. One limitation with the tool is that, for privacy reasons, no unique device identifier is reported per execution. Consequently, if the tool fails to remove a unwanted software strain and a user re-runs the tool, they will be counted twice. This may cause us to overestimate the number of infections per family. As the Chrome Cleanup Tool is opt-in, we caution its metrics cover only a subset of all infected machines. While this precludes absolute estimates on the number of unwanted software installs, we can still estimate the relative population of each software family.

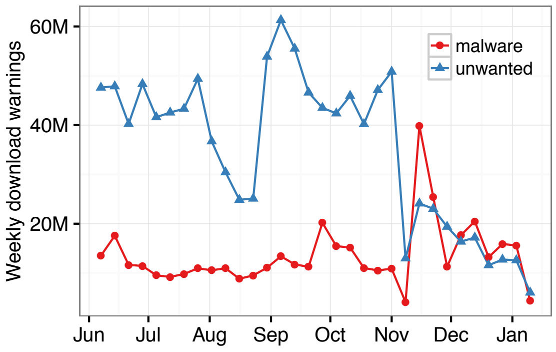


Figure 14: Breakdown of weekly download warnings displayed by Safe Browsing for unwanted software compared to malware. The bursty behavior results from evasion on the part of PPI publishers.

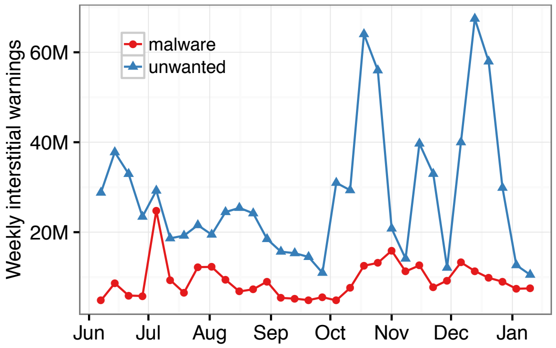


Figure 15: Breakdown of weekly page-level interstitials displayed by Safe Browsing for unwanted software compared to malware. The bursty behavior results from evasion on the part of PPI publishers.

Over the last year, the Chrome Cleanup Tool identified *tens of millions* of installations of unwanted software. We present the top 15 most popular strains flagged from January 8, 2015–January 7, 2016 in Table 6. We measure popularity as the total installs per family divided by all known unwanted software installs. To map these families back to the PPI ecosystem, we mark each family known to distribute via any of the four PPI networks we monitor. We arrive at this determination by running the Chrome Cleanup Tool at the completion of the binary execution phase of our analysis pipeline (described in Section 3) to see whether the tool flagged any of the binary's components.

Country	Frac. Downloads	Price per install
India	8.2%	$0.09
Brazil	7.2%	$0.13
Vietnam	6.4%	$0.06
United States	6.2%	$1.50
Turkey	5.1%	$0.11
Thailand	3.3%	$0.11
Pakistan	3.2%	$0.08
Mexico	2.6%	$0.07
Indonesia	2.5%	$0.09
Philippines	2.5%	$0.08

Table 5: Top 10 countries receiving the largest volume of Safe Browsing warnings related to unwanted software.

Our results indicate that 14 of the top 15 software families flagged by the Chrome Cleanup Tool simultaneously pay for installs during our monitoring. Conduit, the top family, is a browser settings hijacker that accounts for 20.9% of all unwanted software installs reported by the Chrome Cleanup Tool. Multiplug, the most popular ad injector, accounts for 5.1% of installs. Our results illustrate the virulent affect that unwanted software found in the PPI ecosystem has on end users. We caution we cannot definitively say all of these installs stem from PPI[2]; there are potentially other sources of installs such as direct downloads via deceptive websites and advertising. However, paired with millions of Safe Browsing warnings for PPI downloaders, we argue that PPI plays a substantial role in unwanted software installation levels.

6 Distribution Techniques

We conclude our investigation with an examination of the affiliates responsible for distributing PPI downloaders, the landing pages they operate, and the deceptive practices that they employ to drive installs.

6.1 Estimating PPI Affiliates

We estimate the number of affiliates participating in Amonetize, InstallMonetizer, OpenCandy, and Outbrowse by scanning for publisher identifiers that each PPI downloader embeds in offer requests for accounting purposes. Based on the dynamic traces of roughly 1.5 million PPI downloaders provided by Safe Browsing (discussed previously in Section 5), we estimate there are 2,518 publishers in the ecosystem, some of which may participate in multiple PPI networks and thus should not be considered unique. We provide a breakdown per PPI network in Table 7. Drawing these estimates into the broader context of PPI, we find a relatively small ecosystem that consists of hundreds of advertisers paying for unwanted software installs that a few thousand publishers distribute. Despite the low number of actors in the space, the end result is still millions of unwanted download attempts on a weekly basis.

Unwanted Family	Popularity	PPI Advertiser
Conduit	20.9%	✓
Elex	13.4%	✓
Multiplug	5.1%	✓
Crossrider	4.6%	✓
Browsefox	3.8%	✓
My PC Backup	2.8%	✓
Systweak	2.8%	✓
Mobogenie	2.4%	✓
Smartbar	2.2%	✓
Wajam	1.8%	✓
AnyProtect	1.7%	✓
WinZipper	1.5%	✗
Vopackage	1.2%	✓
ShopperPro	1.2%	✓
Vitruvian	1.1%	✓
Other families	33.5%	–

Table 6: Top 15 software families as detected by the Chrome Cleanup Tool on Windows systems. Popularity is the fraction of all known unwanted software installs.

6.2 Landing Pages

In order to drive installs, PPI affiliates must present content that either entices or deceives a victim into downloading and executing a PPI downloader. We obtain a sample of these *landing pages* from Safe Browsing which monitors the entire redirect chain behind unwanted software delivery [27]. However, for privacy reasons, our analysis is restricted to a two week period after which these fine-grained details disappear. In total, we sample the top 15,000 most visited landing pages from January 18–February 1, 2016 that direct to one of the four PPI downloaders we monitor. The sites topping this list include large software companies like *utorrent.com*, *bittorrent.com*, and *savefrom.com* (a YouTube downloading service); download portals like *filehippo.com*; and video and media torrent sites like *thepiratebay.se* that display deceptive ads that in fact link to PPI downloaders.

In order to gain a perspective of the category of sites involved in PPI distribution, we crawl all of the landing pages in our sample and supply the non-HTML formatted text to a topic modeling algorithm similar to Gensim's implementation of LDA [13]. We present the top 10 topics in Table 8, covering 53.6% of all sampled land-

[2] Once a PPI downloader executes, only symptoms related to the bundled advertiser software subsist after the installer completes. The Chrome Cleanup Tool cannot provide us any details for whether unwanted software originated from a PPI downloader.

PPI Network	Binary Samples	Affiliates
Outbrowse	1,182,910	1,106
Amonetize	237,660	420
OpenCandy	43,677	747
InstallMonetizer	22,879	245
Total	1,487,126	2,518

Table 7: Estimate of unique affiliates per PPI network. These affiliates drive millions of weekly downloads to PPI networks.

ing pages. Users searching for freeware, video games, torrents, cracks, and even anti-virus are highly likely to encounter PPI downloaders. Most of these sites (58%) cater to an English audience, followed in popularity by Russian (10%). Our results illustrate that popular download portals (or their contributors) fuel a large segment of unwanted software distribution, in turn receiving a kickback from PPI networks.

6.3 Distribution Pages

After a victim engages with a landing page, PPI affiliates redirect the victim to a *distribution page* that hosts the PPI downloader. This site may be operated by the affiliate or directly by the PPI network, with flavors varying per PPI network. We find that PPI operators rapidly churn through distribution pages, likely to avoid unwanted software warnings from Safe Browsing due to an increasingly negative reputation. During the eight months from June 1, 2015–January 7, 2016, we observed 191,372 distribution pages involved in hosting PPI downloaders. We estimate the lifetime of these pages by measuring the time between the first client that reports a download attempt to Safe Browsing and the last reported download attempt, irrespective of Safe Browsing raising a warning. We find the median lifetime of an Amonetize distribution page is 7 hours, compared to 0.75 hours for Outbrowse. These two stand in contrast to InstallMonetizer and OpenCandy, where distribution pages remain operational for a median of 152 days and 220 days (the entire monitoring window) respectively. This longer lifetime results in part from Safe Browsing not warning on all OpenCandy installs as they do not fall under Google's unwanted software policy, and in part due to Outbrowse and Amonetize controlling distribution pages, simplifying the process of churning through domains.

6.4 Evasion & Cloaking

Even if PPI operators rapidly cycle through distribution pages, there is a risk that Safe Browsing will scan and detect the PPI downloader itself. We find anecdotal evidence that PPI networks work to actively evade this scanning process. For example, when Safe Browsing first launched its unwanted software detection, it covered only executable files. Shortly after, PPI networks switched to distributing .zip compressed binaries to avoid scanning. When Safe Browsing expanded its scanning coverage, PPI networks moved to more esoteric compression formats including .rar and .ace or doubly compressed files. We also observed PPI networks exploiting a limitation in Chrome, where files downloaded through Flash were not subject to Safe Browsing scans. After a recent Chrome patch to address this, PPI networks switched to password protecting their compressed files, providing instructions for victims on how to access the contents. We provide screenshots of each of these techniques in action in the Appendix. This arms race illustrates that PPI networks opt to actively circumvent user protections rather than ceasing to distribute harmful unwanted software. This behavior likely stems from an incentive structure within PPI where remaining profitable entails racing to the bottom of deceptive install tactics.

Site Category	Fraction of Sites
Freeware & Shareware	11.8%
Video Games	10.6%
File Sharing & Hosting	7.3%
Online Video	7.0%
Operating Systems	4.3%
Mobile Apps & Add-Ons	3.7%
Hacking & Cracking	2.7%
Photo & Video Software	2.3%
Game Cheats & Hints	2.1%
Antivirus & Malware	1.9%
Other	46.4%

Table 8: Categorization of the top 15,000 pages driving traffic to PPI downloaders based on topic modeling.

6.5 Promotional Tools

For affiliates that do not operate download portals or peer to peer sharing sites, PPI resellers provide deceptive "promotional tools" that socially engineer web visitors into running PPI downloaders. These tools fall into four flavors: butterbars, ad banners, landing pages, and content unlockers.

Butterbars: PPI resellers like NetCashRevenue provide a JavaScript stub to website operators that generates a yellow bar at the top of a page alerting a victim that their "Flash player is out of date!". This bar can either initiate an auto-download upon visiting the page, or require a victim to click. Either way, the victim receives a PPI downloader.

Content Lockers: Content lockers present victims with an enticing video, song, or PDF. In order to view this content however, a victim must first install a "codec" that is in fact a PPI downloader. Resellers simplify this process by providing a drop-in script that handles spoofing a fake video player and codec alert.

Ad banners & Landing Pages: Resellers will provide webmasters with ad banners or entire customized landing pages that spoof popular software downloads including uTorrent, Java, Flash, and Firefox that are in fact PPI downloaders.

These techniques highlight that even if the software delivered by a PPI downloader appears benign, the distribution practices of affiliates add an additional layer into the determination of whether software is ultimately unwanted. Consequently, advertisers, publishers, and PPI networks all bear responsibility for the current state of commercial pay-per-install and its ties to unwanted software.

7 Related Work

Blackmarket Pay-Per-Install: Our work is influenced in part by prior explorations of the blackmarket pay-per-install ecosystem that sells access to compromised hosts. Industry reports initially qualitatively described these underground markets as early as 2009 [10,33]. Caballero *et al.* performed the first in-depth investigation by infiltrating the markets and tracking the malware families paying for installs [4]. Prices per install ranged from \$0.02–\$0.18, an order of magnitude less than the prices we observed for commercial PPI. These low rates make blackmarket PPI a better bargain for malware distribution over commercial PPI, though evidence exists of cross-over, such as the commercial PPI network iBario recently distributing Sefnit [35]. Other studies have explored the relationships between blackmarket PPI networks and particular malware families [23, 30]. However, all of these studies were limited to establishing a link between the most notorious malware families and their simultaneous distribution in blackmarket PPI; none determined whether PPI was the primary distribution mechanism (as opposed to social engineering or drive-bys). Our study went one step further, establishing the volume of weekly download attempts to commercial PPI downloaders.

Unwanted Software: Unwanted software is not a new threat. In 2004, Saroiu *et al.* found at least 5% of computers connected to the University of Washington's campus network were infected with some form of spyware [32]. In 2005, Edelman tracked multiple purported spyware and adware companies including Claria, WhenU, and 180Solutions to identify their deceptive installation methods and their monetization model [11,12]. More recently, Thomas *et al.* found that 5% of unique IPs accessing Google websites exhibited symptoms of ad injection [34], while Jagpal *et al.* identified millions of browsers laden unwanted extensions performing ad injection, search hijacking, and user tracking [17]. Researchers have also explored some of the distribution techniques involved. In 2006, Moshchuk *et al.* crawled and analyzed 21,200 executables from the Internet and found 13.4% contained spyware [25]. Kammerstetter *et al.* repeated a similar study limited to sites purportedly hosting cracks and key generators, though they found the majority bundled malware, not unwanted software [19]. Our work explored the commercialization of these distribution practices as simplified by commercial pay-per-install affiliate networks.

More recently, Kotzias *et al.* explored code-signing techniques of unwanted software that may lead to reduced detection [21]. We rely on a similar technique for clustering advertiser binaries, though we note that only 58% of the 1,809 unique offer digests we identified contained a signature; similarly, only 50% of 1.5 million PPI downloaders distributed by publishers contained a signature. This may lead to a bias in analysis that focus solely on signed unwanted software. Contemporaneous with our own study, Kotzias *et al.* explored the download graph of unwanted software via Symantec's WINE database and identified 54% of users were affected by unwanted software [20]. Similarly, Nelms *et al.* explored the role of deceptive advertising in enticing victims into running PPI downloaders [26]. Combined with our own work, these three studies present a broad perspective of the number of users affected by unwanted software, an insider perspective of how advertisers, affiliate networks, and publishers coordinate, and the deceptive practices used to entice downloads via advertisements or free software sites.

8 Conclusion

Our work presented the first deep dive into the business practices underpinning the commercial pay-per-install ecosystem that sells access to user systems for prices ranging from \$0.10–\$1.50 per install. Our study illustrated that PPI affiliate networks supported and distributed unwanted software ranging from ad injectors, browser settings hijackers, and system utilities—many of the top families that victims proactively purge from their machines with the aid of the Chrome Cleanup Tool. In aggregate, the PPI ecosystem drove over 60 million weekly download attempts, with tens of million installs detected in the last year. As anti-virus and browsers move to integrate signatures of unwanted software into their malware removal tools and warning systems, we showed evidence that commercial PPI networks actively attempted to evade user protections in order to sustain their business model. These practices demonstrate that

PPI affiliate networks operated with impunity towards the interests of users, relying on a user consent dialogue to justify their actions—though their behaviors may have changed since the conclusion of our study. We hope that by documenting these behaviors the security community will recognize unwanted software as a major threat—one that affects three times as many users as malware.

In response to deceptive behaviors within the commercial PPI ecosystem, members of the anti-virus industry, software platforms, and parties profiting from commercial PPI have formed the Clean Software Alliance [6]. The consortium aims to "champions sustainable, consumer-friendly practices within the software distribution ecosystem." This includes defining industry standards around deceptive web advertisements, user consent, software functionality disclosure, and software uninstallation. These goals reflect a fundamental challenge of protecting users from unwanted software: it takes only one deceptive party in a chain of web advertisements, publishers, affiliate networks, and advertisers for abuse to manifest. It remains to be seen whether the approach taken by the Clean Software Alliance will yield the right balance between software monetization and user advocacy.

Acknowledgments

We thank the Safe Browsing and Chrome Security team for their insightful feedback in the development of our study on unwanted software and pay-per-install. This work was supported in part by the National Science Foundation under grants 1619620 and by a gift from Google. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

[1] AVG. Become an AVG affiliate. http://www.avg.com/affiliate/us-en/become-an-avg-affiliate, 2016.

[2] Ulrich Bayer, Paolo Milani Comparetti, Clemens Hlauschek, Christopher Kruegel, and Engin Kirda. Scalable, behavior-based malware clustering. In *Proceedings of the Network and Distributed System Security Conference*, 2009.

[3] Business Wire. Perion partners with lenovo to create lenovo browser guard. http://www.businesswire.com/news/home/20140618005930/en/Perion-Partners-Lenovo-Create-Lenovo-Browser-Guard, 2014.

[4] Juan Caballero, Chris Grier, Christian Kreibich, and Vern Paxson. Measuring pay-per-install: The commoditization of malware distribution. In *Proceedings of the USENIX Security Symposium*, 2011.

[5] Chrome. Chrome cleanup tool. https://www.google.com/chrome/cleanup-tool/, 2016.

[6] Clean Software Alliance. Sustainable, consumer-friendly practices. http://www.cs-alliance.org/, 2016.

[7] Comodo. Consumer affiliate. https://www.comodo.com/partners/consumer-affiliate.php, 2016.

[8] CrunchBase. InstallMonetizer. https://www.crunchbase.com/organization/installmonetizer#/entity, 2016.

[9] CrunchBase. OpenCandy. https://www.crunchbase.com/product/opencandy#/entity, 2016.

[10] Nishant Doshi, Ashwin Athalye, and Eric Chien. Pay-Per-Install The New Malware Distribution Network. https://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/pay_per_install.pdf, 2010.

[11] Ben Edelman. Claria's misleading installation methods - ezone.com. http://www.benedelman.org/spyware/installations/ezone-claria/, 2005.

[12] Ben Edelman. Pushing spyware through search. http://www.benedelman.org/news/012606-1.html, 2006.

[13] gensim. models.ldamodel – Latent Dirichlet Allocation. https://radimrehurek.com/gensim/models/ldamodel.html, 2015.

[14] Chris Grier, Lucas Ballard, Juan Caballero, Neha Chachra, Christian J Dietrich, Kirill Levchenko, Panayiotis Mavrommatis, Damon McCoy, Antonio Nappa, Andreas Pitsillidis, et al. Manufacturing compromise: the emergence of exploit-as-a-service. In *Proceedings of the Conference on Computer and Communications Security*, 2012.

[15] Orr Hirschauge. Conduit diversifies away from 'download valley'. http://www.wsj.com/articles/SB10001424052702304547704579563281761548844, 2014.

[16] HowToGeek. Here's what happens when you install the top 10 download.com apps. http://www.howtogeek.com/198622/heres-what-happens-when-you-install-the-top-10-download.com-apps/, 2014.

[17] Nav Jagpal, Eric Dingle, Jean-Philippe Gravel, Panayiotis Mavrommatis, Niels Provos, Moheeb Abu Rajab, and Kurt Thomas. Trends and lessons from three years fighting malicious extensions. In *Proceedings of the USENIX Security Symposium*, 2015.

[18] Java. What are the ask toolbars? https://www.java.com/en/download/faq/ask_toolbar.xml, 2015.

[19] Markus Kammerstetter, Christian Platzer, and Gilbert Wondracek. Vanity, cracks and malware: Insights into the anti-copy protection ecosystem. In *Proceedings of the ACM Conference on Computer and Communications Security*, 2012.

[20] Platon Kotzias, Leyla Bilge, and Juan Caballero. Measuring PUP Prevalence and PUP Distribution through Pay-Per-Install Services. In *Proceedings of the USENIX Security Symposium*, 2016.

[21] Platon Kotzias, Srdjan Matic, Richard Rivera, and Juan Caballero. Certified PUP: Abuse in Authenticode Code Signing. In *Proceedings of the 22nd ACM Conference on Computer and Communication Security*, 2015.

[22] Christian Kreibich, Nicholas Weaver, Chris Kanich, Weidong Cui, and Vern Paxson. Gq: Practical containment for measuring modern malware systems. In *Proceedings of the ACM SIGCOM Internet Measurement Conference*, 2011.

[23] Bum Jun Kwon, Jayanta Mondal, Jiyong Jang, Leyla Bilge, and Tudor Dumitras. The Dropper Effect: Insights into Malware Distribution with Downloader Graph Analytics. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, pages 1118–1129, 2015.

[24] LavaSoft. LavaSoft affiliate program. http://affiliates.lavasoft.com/, 2016.

[25] Alexander Moshchuk, Tanya Bragin, Steven D. Gribble, and Henry M. Levy. A crawler-based study of spyware in the web. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2006, San Diego, California, USA*, 2006.

[26] Terry Nelms, Roberto Perdisci, Manos Antonakakis, and Mustaque Ahamad. Towards Measuring and Mitigating Social Engineering Malware Download Attacks. In *Proceedings of the USENIX Security Symposium*, 2016.

[27] Niels Provos. All about safe browsing. http://blog.chromium.org/2012/01/all-about-safe-browsing.html, 2012.

[28] Moheeb Abu Rajab. Year one: progress in the fight against unwanted software. https://googleonlinesecurity.blogspot.com/2015/12/year-one-progress-in-fight-against.html, 2015.

[29] Moheeb Abu Rajab, Lucas Ballard, Noé Lutz, Panayiotis Mavrommatis, and Niels Provos. Camp: Content-agnostic malware protection. In *Proceedings of the Network and Distributed System Security Conference*, 2013.

[30] Christian Rossow, Christian Dietrich, and Herbert Bos. Large-scale analysis of malware downloaders. In *Detection of Intrusions and Malware, and Vulnerability Assessment: 9th International Conference, DIMVA 2012, Heraklion, Crete, Greece, July 26-27, 2012, Revised Selected Papers*, pages 42–61, 2013.

[31] Ben Fox Rubin. Perion sees soaring 2014 earnings following merger. http://www.wsj.com/news/articles/SB10001424052702304815004579417252707242262, 2014.

[32] Stefan Saroiu, Steven D. Gribble, and Henry M. Levy. Measurement and analysis of spywave in a university environment. In *Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation - Volume 1*, NSDI'04, pages 11–11, 2004.

[33] Kevin Stevens. The Underground Economy of the Pay-Per-Install (PPI) Business. http://www.secureworks.com/cyber-threat-intelligence/threats/ppi/, 2009.

[34] Kurt Thomas, Elie Bursztein, Chris Grier, Grant Ho, Nav Jagpal, Alexandros Kapravelos, Damon McCoy, Antonio Nappa, Vern Paxson, Paul Pearce, Niels Provos, and Moheeb Abu Rajab. Ad injection at scale: Assessing deceptive advertisement modifications. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2015.

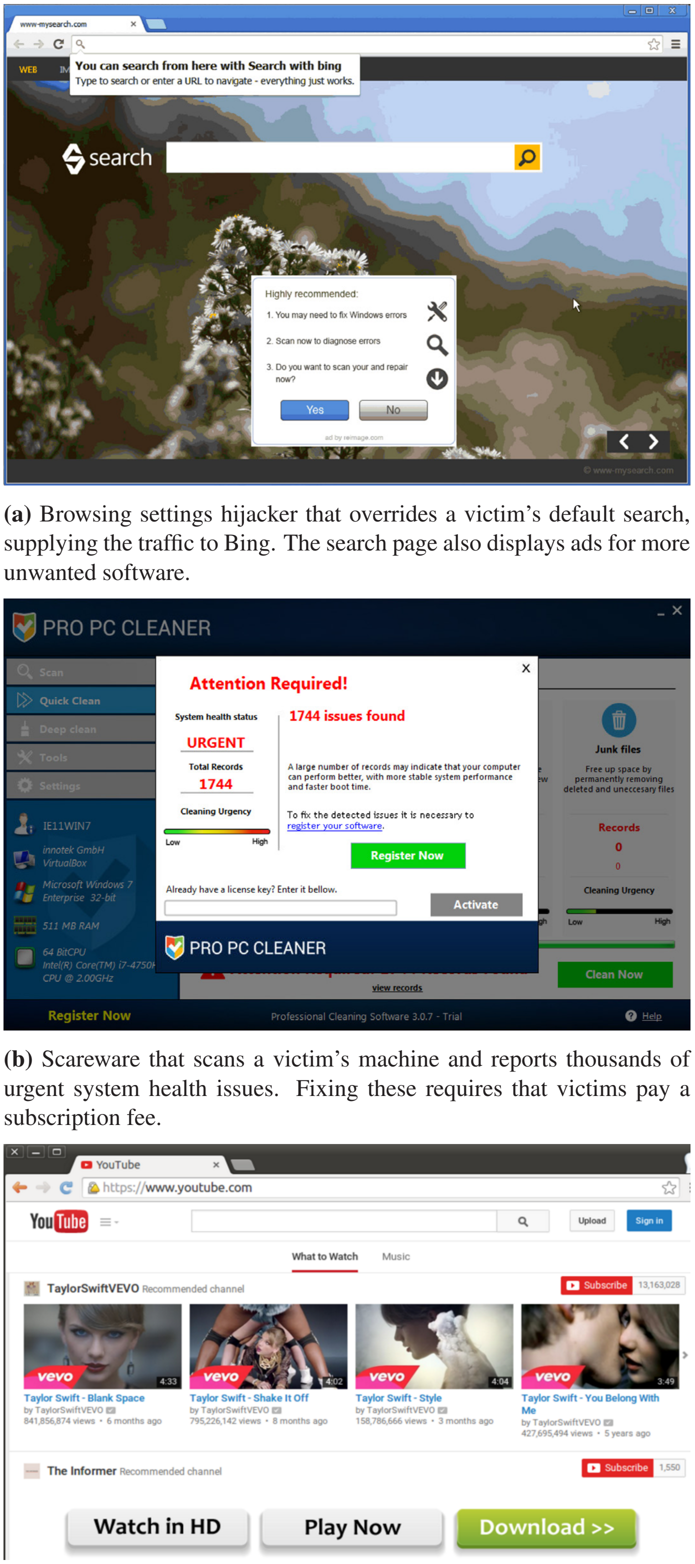
[35] TrendMicro. On the Actors Behind MEVADE/SEFNIT. http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-on-the-actors-behind-mevade-sefnit.pdf, 2014.

[36] VirusTotal. VirusTotal. https://www.virustotal.com/, 2016.

[37] China Internet Watch. Qihoo 360 launched its own affiliate network. http://www.chinainternetwatch.com/7960/qihoo-360-launched-its-own-affiliate-network/, 2014.

[38] Carsten Willems, Thorsten Holz, and Felix Freiling. Toward automated dynamic malware analysis using cwsandbox. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2007.

Appendix



You can search from here with Search with bing
Type to search or enter a URL to navigate - everything just works.

Highly recommended:

1. You may need to fix Windows errors
2. Scan now to diagnose errors
3. Do you want to scan your and repair now?

(a) Browsing settings hijacker that overrides a victim's default search, supplying the traffic to Bing. The search page also displays ads for more unwanted software.

Free up space by permanently removing deleted and uneccesary files

Attention Required!

System health status: URGENT

Total Records: 1744

Cleaning Urgency: Low – High

1744 issues found

A large number of records may indicate that your computer can perform better, with more stable system performance and faster boot time.

To fix the detected issues it is necessary to register your software.

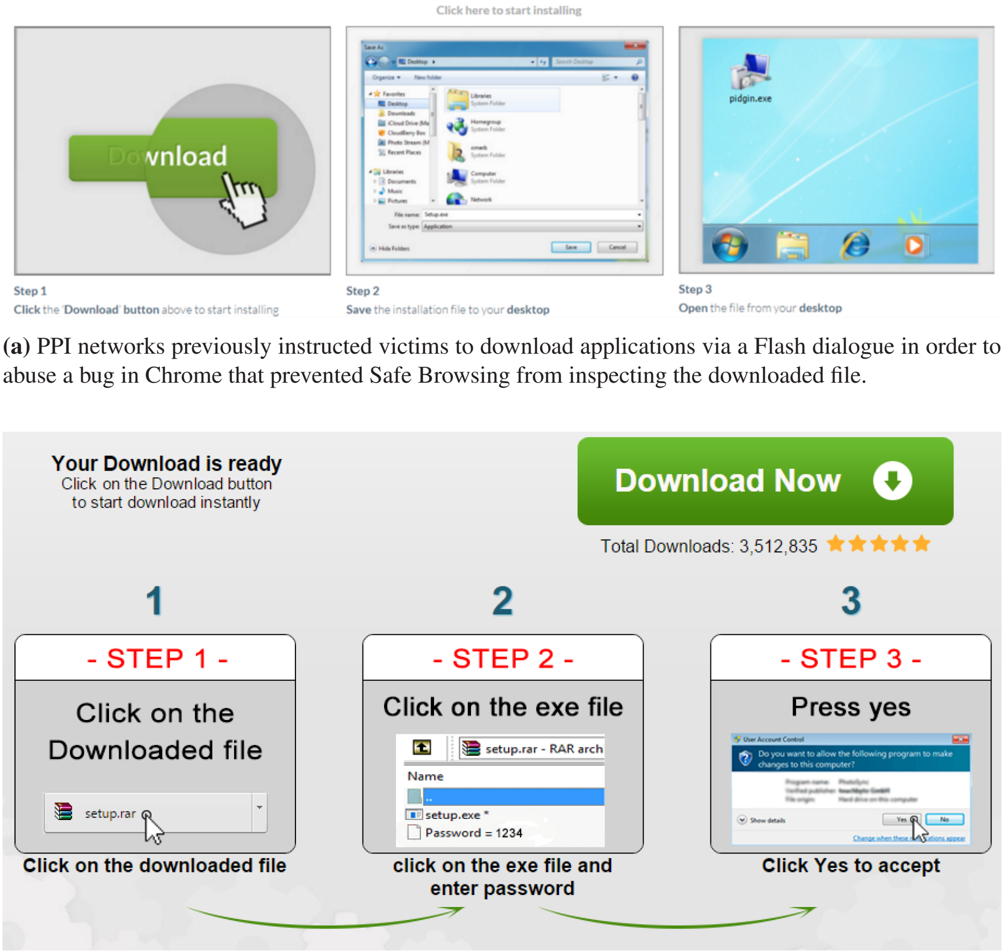
Already have a license key? Enter it bellow.

(b) Scareware that scans a victim's machine and reports thousands of urgent system health issues. Fixing these requires that victims pay a subscription fee.

YouTube
https://www.youtube.com
YouTube
Upload
Sign in
What to Watch
Music
TaylorSwiftVEVO Recommended channel
Subscribe 13,163,028
vevo
4:33
Taylor Swift - Blank Space
by TaylorSwiftVEVO
841,856,874 views · 6 months ago
vevo
4:02
Taylor Swift - Shake It Off
by TaylorSwiftVEVO
795,226,142 views · 8 months ago
vevo
4:04
Taylor Swift - Style
by TaylorSwiftVEVO
158,786,666 views · 3 months ago
vevo
3:49
Taylor Swift - You Belong With Me
by TaylorSwiftVEVO
427,695,494 views · 5 years ago
The Informer Recommended channel
Subscribe 1,550
Watch in HD
Play Now
Download >>

(c) Ad injector that inserts advertisements into pages a victim visits. In this case, the ads direct to more unwanted software.

Sample of user experiences for the software bundled via pay-per-install.


Click here to start installing

Step 1
Click the 'Download' button above to start installing

Step 2
Save the installation file to your desktop

Step 3
Open the file from your desktop

(a) PPI networks previously instructed victims to download applications via a Flash dialogue in order to abuse a bug in Chrome that prevented Safe Browsing from inspecting the downloaded file.

Your Download is ready
Click on the Download button
to start download instantly

Total Downloads: 3,512,835 ★★★★★

1 - STEP 1 -
Click on the Downloaded file

Click on the downloaded file

2 - STEP 2 -
Click on the exe file

click on the exe file and enter password

3 - STEP 3 -
Press yes

User Account Control
Do you want to allow the following program to make changes to this computer?
Show details
Yes
No

Click Yes to accept

(b) PPI network previously instructed victims to download password-protected compressed executables in order to prevent inspection of the downloaded file by Safe Browsing.

Sample of now defunct techniques employed by PPI networks to deliver PPI downloaders while evading Safe Browsing.