AppEsteem™
Certifying apps for a better world

Beta readiness review

Learnings, next steps, request for support

Dennis Batchelder
AppEsteem Corporation
July 2016

AppEsteem provides a safe haven for clean software monetization vendors…

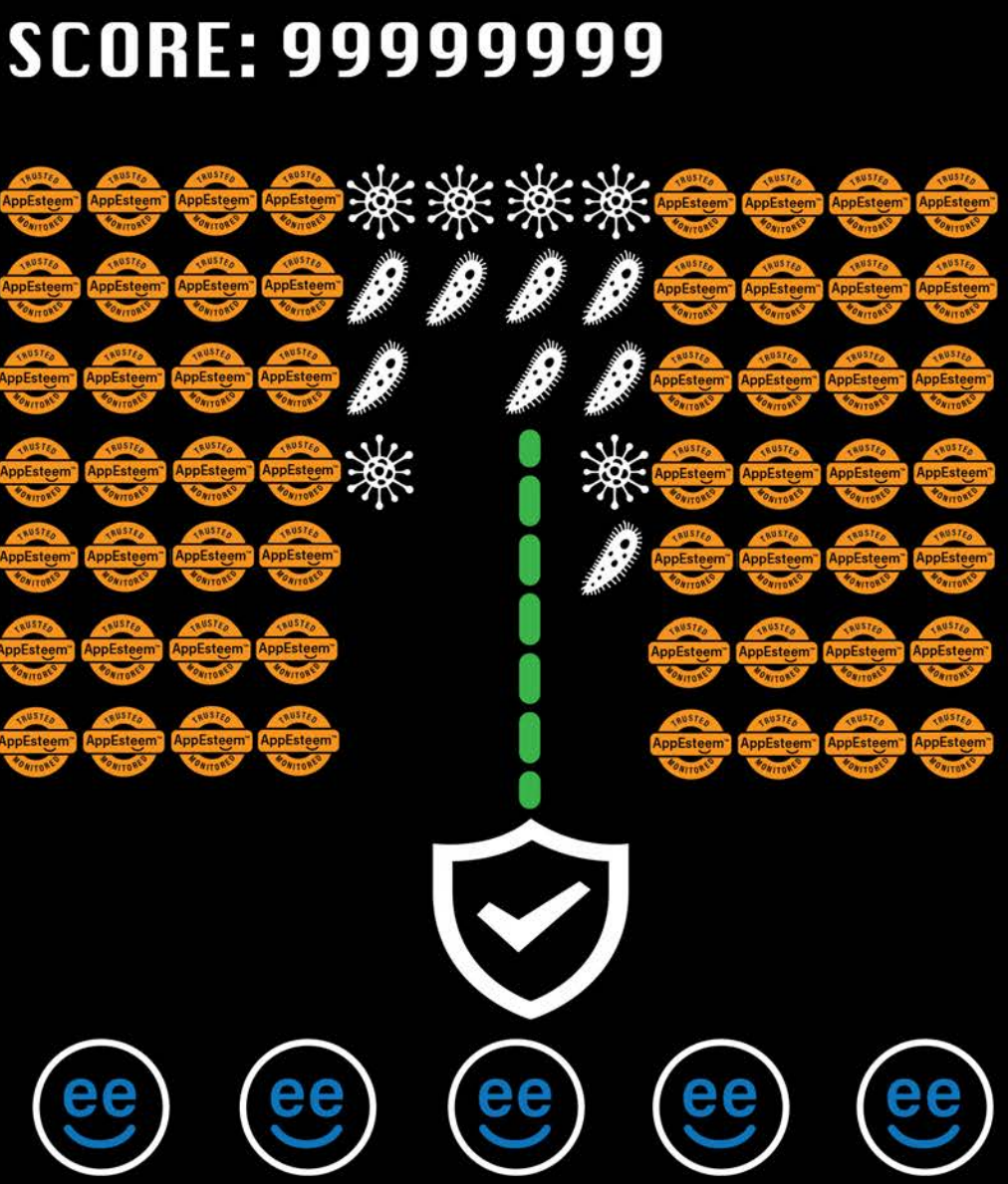
…so we can squeeze out the dirty players

who fund their business by tricking and cheating customers

who grow their business by outbidding the clean players

We help security partners protect their customers from PUA

Before AppEsteem



Certified apps make a better world

Beta Readiness

April - June: Prepare

- ✓ Road show with AVs, platforms, CSA, Compliance officers (5 shows, visits)
- ✓ Recruit dev/research team (12 on team)
- ✓ Sign up installers/downloaders/vendors (2 installers, 3 vendors, 6 CRXs)
- ✓ Figure out monetization plan (fee schedule socialized)

Still in progress…

- Establish validation and certification scorecards (have drafts)
- Deliver first cut of SRCL and seals (SRCL going out next week)
- Get more Security Partners into beta (that's today)
- Land MOU with CSA

August - September: run betas/pilot

- Windows PE (August)
- CRX (September)

How it works

	Vendor	AppEsteem	Compliance Partner	Security Partner	Fee
Sign Up	Provide basic info Sign license agreement May choose compliance partner	Countersign agreement Generate appkey	Accept client		
Monitor	Build and distribute app View telemetry	Construct behavior graph Anonymize, aggregate, publish	View telemetry		
Validate	Submit/renew vendor disclosure every year	Investigate and approve/reject	Sign off on disclosure interview	View approved disclosures	$2000
Certify	Submit/renew certification request every year/every major version	Analyze, test, and approve/reject Construct behavior graph Generate seal	Sign off on certification request	View approved certifications	$200
Integrate	Rebuild with seal Register final package	Re-certify and publish ”signature”		Consume “signatures”	
Monitor	View telemetry	Update behavior graph; alert on anomalies Anonymize, aggregate, publish	View telemetry, alerts	View telemetry, alerts	1% of LTV60
Enforce	Fix app and resubmit	Notify vendor of block Initiate remediation process	Signs off on changes	If blocking: notify why Consume “signatures”	

What the beta (pilot) will measure

Can we increase customer offer satisfaction?	Can we reduce vendor evasion?
***Hypothesis**: Offer screens trick customers to clicking through and leave them dissatisfied. Sealed apps with certified offer screens will lead to better informed and happier customers.*	***Hypothesis**: Today's installers evade detection by morphing hosting locations, digital signatures, product updates, brand names, and domains. Sealed apps won't need this, which reduces the cost to protect*
Measuring success Reduced sealed offer startup rates. Increase sealed app lifetime	**Measuring success** Less evasion: reduced certificates, domains, landing pages, product updates for sealed offers

Validation: where we are

What we've done

- Collected data on our own
- Conducted investigations using public data (Glassdoor, lawsuits, IP ownership)

What we've learned

- Disclosures and vendor commentary seem to be the most appropriate approach
- Compliance partners will help
- Structured interviews will reduce our investigative time

What we plan to gather and make available to security partners

Category	Data
Structural Information	Ownership, DBAs, Addresses, Contacts, Licenses, shared ownership companies
Business Relationships and potential conflicts of interest	Partnerships, Affiliates, trademark disputes, areas not following guidelines
Evidence of Controls	Affiliate management, Advertiser management, IP protection. Supply chain management
Attestations to following clean guidelines	Commitments
(Investigation results)	Public reputation, news, posts

Certification: where we are

Where we started

- Google's Unwanted Software Policies
- MMPC's Objective Criteria
- CSA's Guidelines
- Inherited principles: Consumers need consent, control, and no unpleasant surprises

What we learned

- Missing principle: consumers shouldn't feel cheated after paying
- Important to track the entire creative->landing page->install supply chain

Addressing gaps we've found

Gap	New Requirements
No app monitoring requirement leaves vendors without verification	Apps must link and not evade SRCL library, must honor "uninstall" command
Great apps can still have bad affiliates, causing suspensions by platforms	Landing pages must block obscured references, must publish affiliate restrictions
Normal "next" install flow leaves consumers surprised	Unrelated offers must have unselected radio buttons where the consumer must choose to continue
Need better context to do a fair evaluation	Require apps to submit a value and monetization statement
In-product upgrades need evaluation	System utilities must have a reputable 3rd party vouching for their value
Ad injection has standards too	Set the toolbar bit for AppNexus auctions/equivalent

The seal: where we are

Where we started

- We heard concerns of using Taggants
- We planned to roll our own seal to support our capabilities

What we learned

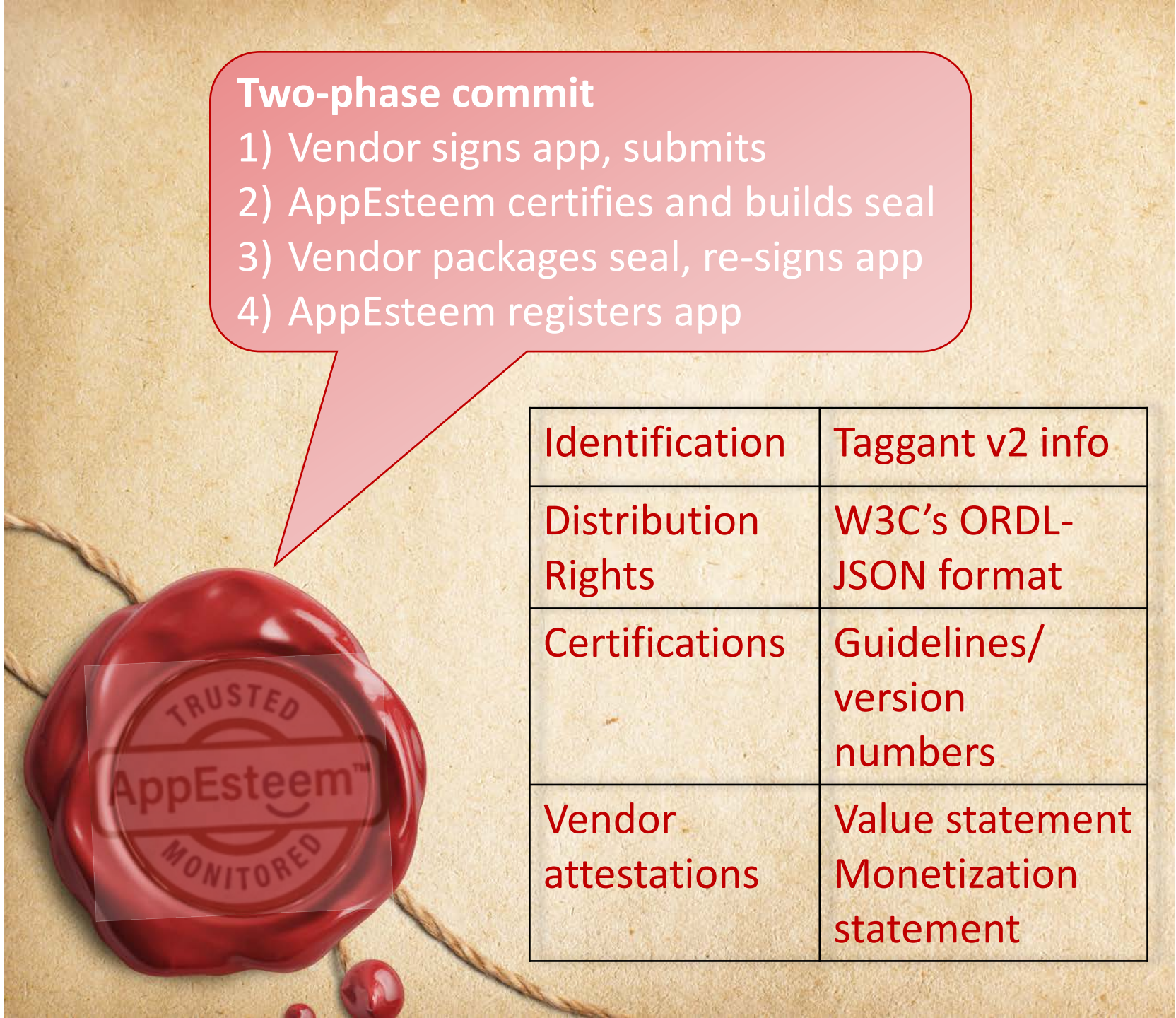
- We need to be open and allow competitors
- We need to reduce implementation friction
- Several AVs already implemented Taggants
- Better to patch holes than introduce brand-new security

Taggant implementation plan

- Single signer (AppEsteem)
- New data inside: distribution rights, certifications, vendor attestations

Two-phase commit

1) Vendor signs app, submits
2) AppEsteem certifies and builds seal
3) Vendor packages seal, re-signs app
4) AppEsteem registers app



Identification	Taggant v2 info
Distribution Rights	W3C's ORDL-JSON format
Certifications	Guidelines/ version numbers
Vendor attestations	Value statement Monetization statement

Monitoring: where we are

What we planned

- Easy linking with our Self Regulating Client Library (SRCL)
- Work with PEs, CRXs, APKs
- Report heartbeat, time to live, blocks, anomalies
- Easy way for Security Partners to report problems

Built for PE files

- Using Microsoft Detours, auto-inject unsealed child processes to monitor registry, file, process, (soon network)
- Screenshot samples to capture offers

Building for CRXs

- CRXs: using AspectJS to monitor
- Screenshot samples to capture ad injection

Building a behavior graph

Category	Data
App Information	• Provenance • Landing page • Identification • Install locations
Components	• Libraries • Children • Parents
Actions	• Processes • Libraries • File, registry, process • Cookies, bookmarks, history, tabs • Default overrides • Advertising

Remediation: where we are

Where we started

- Goal is to encourage the right behavior
- Hope to never need to use the nuclear options

Where we are

- Needs lots of IQ investment to get this right

Remediation thoughts: proportional and escalating response

Stage	Actions
Stop bad behavior immediately	• Security Partners block new installs • Vendor informed of specific reasons • Block new seals from vendor
Demonstrate urgency	• Throttled/targeted removals • Deep investigations
Revoke app	• Full removals
Revoke company	• Full cleanup

Security Partners: time to commit ☺

Register as a Security Partner

- Security Partner access is FREE
- http://appesteem.com -> REGISTER
- Sign our partnership agreement (we'll send out next week)

What you get

- Validated company and sealed certification disclosures
- Access to sealed apps and analysis results
- Distribution and behavior telemetry
- Signatures and online checks

During Beta: pre-sign off

- We're learning this together: we want to get it right
- Validation and Certification disclosures
  - We want to pivot as necessary
- Every install package
  - We want to ensure our behavior graphs are complete
- Work to help us get remediation right
  - We want to put serious pressure on the bad guys

AppEsteem™

Certifying apps for a better world

http://appesteem.com
info@appesteem.com
@appesteem