Before the
Federal Communications Commission
Washington, D.C. 20554

In the Matter of	)	
	)	
Modernization of the Nation's Alerting Systems	)	PS Docket No. 25-224
	)	
	)	
Wireless Emergency Alerts	)	PS Docket No. 15-91
	)	
Amendment of Part 11 of the Commission's Rules Regarding the Emergency Alert System	)	PS Docket No. 15-94
	)	

Comments of Sage Alerting Systems, Inc.
Regarding the Modernization of the Nation's Alerting Systems FNPRM

August 31, 2026

1. Introduction

Sage Alerting Systems has been actively involved in the Emergency Alert System (EAS) since its inception 32 years ago. We are pleased that the FCC is now directly addressing what has been called "Software EAS", for lack of a better name – a set of rules changes that will move the implementation of EAS out of its 1994 custom hardware box solution and into a more modern era of EAS functions distributed across the components of a broadcast facility, improving security, performance, redundancy, and manageability.

2. Discussion of Software EAS

Sage has developed and demonstrated a prototype Software EAS system. In doing so, we have necessarily revisited many aspects of the current EAS specifications, protocols, and procedures. We have also reviewed cybersecurity aspects of protecting the Software EAS system and the platform on which it runs.

We have considered four types of Software EAS:

1) Software running on a single-application platform, but using simplified hardware, meaning virtual ports, remote displays and indicators, AoIP[1] with optional hardware interfaces for analog audio. The EAS vendor is responsible for producing software update packages that are installed by the end user, as is common for standalone EAS today.
2) Software running on a closed multi-application platform that is produced and tested by a non-EAS broadcast equipment manufacturer. All of the platform's software and addons like watermarking and meta data encoding have been tested for interoperability and security, and the installation of software on the hardware is controlled by that manufacturer. The Software EAS package is one of the optional software components that can be installed on that hardware. The hardware manufacturer and the Software EAS provider work together to integrate, certify, and distribute the Software EAS component and updates to it for secured installation by the end user.
3) Software running on hardware built for broadcast use and deployed at several facilities owned by a single owner, but the software stack is controlled by an in-house software integrator that handles security and interoperability. Software EAS is added to that stack.
4) Software EAS is sold as a downloadable package where the end user is responsible for most security and interoperability aspects. Sage does not recommend this approach.

   Other types are possible, the differentiation is that acceptable systems involve coordination between the platform provider and the Software EAS provider, where configurations are controlled and security updates can be provided. Unacceptable types are instances where the end user is left to set up, configure, and maintain a disparate set

[1] Audio over IP, a broad definition encompassing a variety of audio formats and transport mechanisms. Our system has been demonstrated using streaming protocols such as AAC over HTTPS, and AES67. All of these are in common use by broadcasters.

of modules with no assistance or interoperability testing and no manufacturer support for security and secure updates.

Sage believes that the first three types of Software EAS can meet the existing Part 11 requirements for EAS systems with the changes proposed by the FCC in the FNPRM to allow virtualized interfaces such as audio ports, indicators, and displays.

The FNPRM asks questions about certification and additional testing for Software EAS devices. Briefly, Sage says that protocols and procedures that make up EAS are the same for both standalone and Software EAS systems. While additional testing based on changes made to EAS should be considered as we move forward, Software EAS should not be required to pass tests that are not required for standalone systems. We expand on this later in our comments.

The FNPRM asks about certification for Software EAS regarding the 2011 FEMA IPAWS Conformity Assessment (CA) Program. We agree that those items should be tested – Software EAS should be held to the same software technical standards as standalone systems.

The status of the CA program is unknown. It may be that this test program is not currently available, and that it cannot be revived in a timely manner. We detail existing precedent in EAS history where a rigorous test plan performed by the manufacturer and reviewed by a Telecommunications Certification Body (TCB) resulted in Part 11 certifications.

Insistence on an active CA program halts new standalone devices as well as new Software EAS systems. A suitable workaround exists for both cases.

The FCC proposes a twelve-month delay for Software EAS systems. This is unnecessary, especially for the first three examples above for tightly administered Software EAS systems.

3. Comments on specific questions asked in the FNPRM

Numbers in parentheses in the headers of the following sections refer to the FNPRM paragraph numbers.

3.1 Digital Signatures for CAP messages (41-44)

The FCC proposes to require that CAP formatted EAS messages have a valid digital signature. We agree. Also, Part 11 should include an explicit definition of what a valid signature means in CAP/EAS context. We suggest:

> A valid digital signature meets the requirements of CAP 1.2[2]
>
> and
>
> a chain of trust is established between the alert originator's signing certificate included in the alert's XML data and the trust anchor in the EAS device, in which none of the certificates are expired,
>
> and

[2] OASIS Open, *Common Alerting Protocol Version 1.2 (July 1, 2010)*, section 3.3.4.1 Digital Signatures https://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html

the document's cryptographic signature is correct - that is, the digest in the signature matches the digest computed by the receiver, and the signed digest was signed by the certificate included in the signature.

Sage believes that a valid digital signature should be required for any CAP message intended for relay by an EAS system. The signature should rely on intermediates and roots that can be pre-downloaded, so that the EAS device does not need real-time access to the internet to download certs on demand. This allows use of one-way satellite or other distribution schemes to send alerts, providing alternative sources of alerts when the internet, or IPAWS Open, is unavailable.

The exact method of providing key distribution in these cases then becomes a matter of coordination between alert distribution systems and EAS device manufacturers to keep keys up to date. Users must also keep their systems up to date by installing updates provided by device manufacturers.

Para 43

Many originators that we are aware of use FEMA IPAWS as their distribution method for CAP EAS messages. They use a variety of software origination packages. It is technically possible for originators to use a non-IPAWS aggregation and distribution server, while signing their alerts with their IPAWS signing certificates. They could send the message via both IPAWS and other distribution systems. When received, no matter the distribution path, EAS equipment would be able to validate that the message came from an IPAWS certified originator. This would reduce the key management burden for EAS users, if IPAWS signing certificates are permitted for use on non-IPAWS servers.

Para 44

Paragraph 44 discusses CAP alerts and possible delayed or rejected messages, though the footnote refers to a discussion in the CSRIC VI EAS Security Report of adding signatures to legacy EAS messages. For CAP messages, there were early growing pains with the use of signatures, especially with the translation of extended character sets in message transport causing signature validation errors. There have been no such problems since 2021.[3]

As to the use of digital signatures in EAS alerts, Sage participated in the writing of the CSRIC VI WG 2's security report, and the information provided in that report's section 6.1.5.2 is as valid today as it was in 2018[4]. We discuss digital signatures for legacy below.

3.2 Digital Signatures for Legacy EAS alerts (45-48)

Signing a legacy EAS message is straightforward and can use well-known techniques for detached signatures. Adding signatures to legacy EAS alerts will add some complexity, fragility, and minor delays in alert handling; NWR SAME protocol interoperability issues will also be a concern.

---

[3] Sage independently logs all alerts from the IPAWS feed for debugging and user support. We have found that FEMA's IPAWS Open has a high degree of reliability, and the digital signature processing does not add any unnecessary delays or add additional risk.

[4] CSRIC VI, *Final Report—Security Aspects of Emergency Alerting System (EAS)* at 27, 33-34 (2018), https://www.fcc.gov/file/14853/download (CSRIC VI EAS Security Report).

Here is one way to add signatures to legacy EAS:

1. Send signature frames before a ZCZC legacy message is sent. This would include at least an ID of the public key needed to verify the signature, and the signed hash. An ID is sent rather than the full public key to reduce the size of the signature frame. The length of the signature frame needs to be as short as possible, but signatures will take a few seconds each, and for redundancy must be sent more than once, as are the current headers.
2. EAS devices would need to periodically access an online database of IDs and Certificates, download, and store them. This is to permit key verification even if the internet is not available at the time the signed legacy message is received. This database is a signed trusted source and must be administered by someone - FEMA is one possibility.
3. Signature frames must be sent before the legacy message, to allow the legacy message headers to be verified and the alert to be relayed while it is in progress (as with an EAN). The signature frames must be formatted for decoding by updated devices while not confusing unmodified devices.
4. Only originators of legacy messages (other than RWT) would need private keys. In the current rules, anyone can originate alerts. As a practical matter, comparatively few broadcast stations need to originate (LP1 and LP2), and even fewer are equipped with live staff and the training necessary to do so. Limiting the number of key holders simplifies key management.
5. An EAS participant relaying a legacy alert only needs to relay the received signature along with the alert. It is not necessary to generate a new signature. The signature would not include the LLLLLLLL (station identifier) portion of the alert, which is the only field that changes during a relay.
6. To solve the problem of signing a legacy alert that was received from CAP, the CAP alert must include the contents of the legacy signature frame. The process of converting a CAP message to a legacy message is well documented, as this is one of the main purposes of the EGIC Implementation Guide and is necessary to the process of duplicate detection. A CAP message intended for transmission by legacy EAS would need to contain the ZCZC constructed according to the ECIG rules, a hash of that string, and the signed hash. That data is then used by the EAS device to build the signature frame using the ZCZC's signed hash from the CAP message to include in the legacy signature header.
7. On receipt of the signature frames and ZCZC strings, the receiving device computes the hash and uses the ID to find the pre-stored public key. The public key is used to decrypt the signed hash. The decrypted hash is compared to the just-computed hash. If there is a match, the alert is accepted. Note that if any single signature frame matches any single ZCZC frame, it is very likely that a valid frame has been found, improving reception of noisy data. However, if no signature frames were correctly received, then the alert is rejected, even if the alert frames were received correctly - this is a cost of requiring signed legacy EAS.
8. RWT messages need not be signed, as they are never relayed. A Part 11 rule would need to be added to enforce that concept – RWT messages cannot be relayed.

This method improves the security of legacy EAS. The costs are longer transmission times and added risk of missing an alert due to transmission noise. Other methods are possible.

EAS devices that can receive CAP already have the cryptographic tools needed to perform the signature manipulations; they are essentially the same as needed to verify a CAP message. Only devices that need to originate legacy messages need private keys and the ability to sign.

Now that we have shown that digital signing of legacy messages can be done, is it practical?

One issue is NOAA Weather Radio – this is the major source of legacy EAS messages. Unless these messages are also digitally signed, EAS devices would need to accept unsigned legacy messages. The overall EAS system would gain little benefit from the effort and expense to protect some but not all messages and would not gain protection against spoofing.

If the NWS starts sending CAP messages for EAS to IPAWS, the risk is reduced, as legacy messages could be verified against CAP messages, but that requires a constant internet connection. One reason to maintain the legacy EAS system is as a redundant path when there is no internet access. True protection of legacy EAS can only be achieved only when every originator of legacy messages has a unique signing key. If certificates needed to build a chain of trust are periodically downloaded and stored, signed legacy messages can be verified with only a small additional risk. Any scheme that requires internet access when the legacy alert is received will fail when the internet connection is down. If legacy EAS requires continuous internet access, it might as well simply be replaced by a CAP-only system.

Legacy messages would be delayed by the length of the signature frame times the number of repetitions of the frame. The length of the frame depends on the length of the key used. An estimate of 10 to 20 seconds is a reasonable starting point. Sage has tested a system like that described above, using elliptic curve cryptography for smaller keys. The slow data rate of EAS will eventually be overwhelmed by the need for longer keys in the future.

Note that the digital signature of EAS headers protects only the alert header – originator, event, locations, start time and duration. It does not protect the audio content.

Sage believes that it should be possible to design signature headers (and other additional data such as four digit years and alert ID) that can be sent detached from the common ZCZC string. This will require testing of a wide range of devices, and participation of consumer electronics manufacturers that are not tightly coupled to the EAS ecosystem. Most of the EAS systems in the field that are compliant with EAS in 2026 could be updated, though this will need investigation. We should always avoid obsoleting older equipment where possible, especially for a mandate like EAS. We can't halt all progress, however.

3.3 Universal Alert Message ID for EAS (55)

The main impetus for the use of a Message ID, called a data tag in CSRIC VII[5], was to solve the duplicate detection issue in NWS where internal messages sent to multiple transmitters with many FIPS codes were modified at each transmitter to send only the FIPS codes of interest to that particular transmitter. This caused a single weather message to generate several unique NWR EAS/SAME messages, which were not detectable as duplicates if two or more messages entered the EAS relay mesh. Likewise, an NWS CAP message would not necessarily match the

[5] Communications Security, Reliability, And Interoperability Council VII, *Report on Recommendations to Resolve Duplicate National Weather Service Alerts*, https://www.fcc.gov/file/20608/download, Section 6.1

locally received legacy EAS message. Adding a unique message ID would serve to disambiguate the EAS messages.

CSRIC VII rejected modifying the ZCZC string as being too disruptive to legacy EAS and NOAA weather radio[6]. They recommended adding a new EAS frame type outside of the existing header[7]. This is similar to Sage's signature scheme discussed above.

An alternate solution to the duplication issue was in section 6.6 of the same CSRIC report, called "NWS NOAA Weather Radio Sites provide Complete and Consistent Ordering of SAME Location (Local Area Codes) in All Alert Broadcast Messages". This required no changes to the EAS/SAME protocols but did require greater effort by NWS. We believe NWS is making changes similar to what was suggested in 6.6.

3.4 Incentivizing EAS Geofencing (70)

Yes, reducing over-alerting is a good thing. However, EAS over broadcast Radio and TV is a blunt instrument. EAS alerts are heard or seen by the entire audience across the broadcaster's coverage area, regardless of the precision of CAP polygons or FIPS codes. This reduces the effectiveness of Geofencing. Newer technology and smarter user devices can reduce the number of messages displayed by smart phones, smart TVs, and smart radios/digital dashboards, but EAS is sent on the main audio channel - if you are watching or listening to that main channel, you will hear the alert. The only choice for the broadcaster is to send the alert to either all of their audience or none. Currently, polygons and circles are only available in the CAP message, not in legacy EAS.

Schemes suggested over the years to add polygons and circles to legacy EAS require new data tag frame types; modifying the EAS/SAME ZCZC frame is not viewed as viable (see discussion of message ID, above).

If a legacy EAS message is received, and a matching CAP message is found, a rules change would allow an EAS participant to make the relay/no relay choice based on polygons in the CAP message rather than FIPS codes. If a CAP message is not found, only FIPS codes can be used.

If a data tag frame type is defined, legacy EAS could acquire polygon capability. This adds to the length of the header portion of the alert and would need to be included in the digital signature. This adds to the overall complexity of legacy EAS.

Is it worth it? In some cases, yes, particularly in large counties. It would also be useful for broadcasters with a very small coverage area. The effectiveness of geocoding in broadcast depends on local geography, the area covered by a FIPS code, state plans, and training for alert originators so that their expectations can align with how broadcast EAS works. At present, legacy EAS reception often results in wider coverage than expected by the alert originator.

There is an additional consideration for broadcasters that provide the upper levels of the relay chain/mesh. If an LP1, for example, uses polygons to make relay decisions, they might not relay an alert that a monitoring station further down the chain needs to hear. State Plans will need to be reviewed to ensure that monitor assignments are still valid when polygons are used.

---

[6] Id. Section 6.1.1.1

[7] Id. Section 6.1.1.2

Sage agrees that the FCC should permit but not require the use of CAP polygons for filtering. Polygons should not be used for NPT or EAN alerts.

3.5 Improving Target Area (73-75)

Some areas of the country, with the active encouragement of the NWS, have begun using the subdivision extra digit in FIPS codes. This can lead to confusion, as the names used to build the text string for subdivisions are defined in Part 11.31. Should the FCC permit, EAS system manufacturers could allow users in each affected area to provide new names for those subdivision codes.

CAP allows text descriptions that are more accurate, but Part 11 and the EGIC IG still require a formatted string built from the FIPS code, per Part 11.51(d) and ECIG IG 3.6.3 and 3.6.4. This string is then used for the text crawl and text-to-speech generation.

For the subdivision to be of practical utility, it could be treated as a 6th digit of the FIPS code, so that a new area name could be used, and not simply [division of] county. For example, 012087 would be translated to "Monroe County" by the EAS device, while 112087 could be "Key West". This could be left to the SECC and the state plan, in conjunction with local authorities and the NWS, to use a term acceptable to the local population.

This raises the question: does the PSSCCC code in Part 11 use ANSI INCITS 31-2009, or does it become a code that was once FIPS but is now an FCC-defined and regulated code with no formal relationship to an ANSI standard?

The FCC does not currently manage the official list of codes in ANSI INCITS 31-2009. The NWS also uses PSSCCC codes, their version and the FCC's version need to be in sync.

The current definition of NWS SAME, Part 11, the IPAWS profile, and ECIG IG all refer to the PSSCCC code as digits. Using a letter for one of these would be a major change, possibly breaking compatibility with hardware that validates the ZCZC PSSCCC string for digits. Sage does not recommend using letters in the PSSCCC field.

3.6 Symbols for emergency events (77)

Adding graphics to the existing text crawl or display banner brings "character generators" into the mix, and a requirement to add icons could significantly raise the cost of getting crawls on the air for smaller stations. This should be permitted but not required.

3.7 Flexibility (96)

Sage agrees with the FCC that flexibility is one of the most important aspects of its proposal. We don't yet know which of the many possible permutations of a distributed EAS will be the most popular, easiest to produce, most cost effective to maintain, and best integrated into the user's workflow. Over-specification in Part 11 won't spur innovation. It should be permissible for EAS to be concentrated in a single device or spread across multiple devices.

3.8 Service Location (98)

We agree that it is important to maintain the ability to receive EAS alerts from "in market" sources, and to be able to get EAS alerts on the air if a distant program source is disrupted. Software EAS does not fundamentally change how broadcasters can route EAS audio, just the ease with which it is accomplished. Stations use the internet or private networks to receive EAS

audio now, just as they sometimes use STLs with private IP or even the internet for a main or backup link.

We agree that the hardware running the software components of EAS should be under the user's direct control (possibly shared per 11.52). We agree that control of EAS, and responsibility for compliance, must remain with the station licensee. Audio must not be sent to an offsite third-party EAS aggregation site, where alerts are inserted and shipped back for broadcast. In all other respects, the new rules for software EAS should be no more restrictive than the current rules for standalone EAS.

Likewise, the new rules should continue to permit a remote network operations center to provide remote studio capabilities in an emergency, as has been done many times in disasters that flooded local studios or limited access to transmitters. EAS, especially CAP EAS, can be received remotely and then sent to the distressed location, possibly through direct control of the transmitter audio feed or in conjunction with a Software EAS component.

3.9 Additional Limitations (99)

Software EAS does not change the resilience of EAS. While it is true in today's world that the loss of internet connectivity can cause the loss of audio on a station, software EAS does not make this worse. The current Standalone EAS rules do not prohibit the use of streamed EAS audio inputs. Many stations already use various IP transport solutions to get monitor assignment audio to the EAS device, and many of those solutions traverse the public internet. Software EAS might make such IP links more secure by using asymmetric keys to secure remote audio, while increasing resiliency by supporting failover between links, and continuing to support traditional direct-wired audio. The recent R&O reminded stations that they need to make their external connections secure; a new R&O should remind stations that continued access to EAS monitor assignments and the output transmitter are critical as well.

3.10 Supply chain risk (102)

Currently, certified EAS hardware devices are provided by a very small number of companies. These devices are long-lived, the replacement rate is low, and the number of new stations each year is limited. Allowing EAS to run on other platforms is one way to reduce the current reliance of EAS on custom hardware from a limited number of sources.

Sage has stated elsewhere that it plans to offer standalone hardware if the Part 11 rules permit simpler hardware, where the displays, audio interfaces, GPIO, serial ports, and other virtualized components are not mandated to be a part of the hardware.

3.11 Patents (103)

Patents are always a concern; this is true of any technology. Sage has no patents. Software EAS does not involve new technology, just a new regulatory environment. EAS has always had a large software component, and has leveraged browser interfaces, remote components, crypto, IP, codecs, open-source libraries, Linux, Windows and all the rest.

If any commenters are aware of specific issues, or hold relevant patents, they should say so.

3.12 EAS Software Certification (104)

Sage agrees that certification is an important aspect of Software EAS.

The FCC notes that a commenter to the original NAB petition observed, regarding certification, "[w]ithout such a framework, software solutions could vary significantly in quality, security, and functionality, with no clear method for regulators or users to determine compliance."

There is little backing for an uncontrolled free-for-all uncertified Software EAS. Sage does not support that model.

Sage believes that software certification should be required and should align with what is currently required for standalone devices. As for testing software compliance with Part 11 and for proper CAP/EAS interaction defined in the ECIG Implementation Guide, the EAS and CAP protocols are the same, the testing methodology is the same, and the test requirements should be the same. As we discuss below, the existing testing model allows for some amount of in-house "self" testing. If the formal testing offered in 2011 is not available, in-house testing using the same test criteria should be permitted, with complete and substantial documentation submitted to the TCB.

3.13 Current testing model (105-106)

The FCC says that "We tentatively conclude that requiring EAS software to be certified under a conformity assessment scheme that is architecturally similar or identical to that under which EAS equipment is certified today is necessary to preserve the reliability of EAS, i.e., EAS software would be submitted to an approved entity for testing and then to a certification body for approval."

Sage agrees with the FCC's description of current test procedures, though we note that some portions of device testing have not always been performed by an accredited FCC-recognized test laboratory. Self-testing, especially for EAS protocol and procedures, has been permitted in the past. In reviewing EAS device test reports[8] from 2011 and later, we found cases in which EAS testing was performed by the manufacturer's own staff, in the manufacturer's own facility. Test reports, some quite detailed, along with Part 15 tests, were sent to a TCB that issued the Part 11 certification. The TCB did not run the software tests themselves.

An excellent example of rigorous self-testing of EAS formats (header testing) and procedures (duplicate, EAN priority) can be found in the certification test report for the Trilithic EASyCAP1[9].

3.14 What should be tested (107)

In brief, Software EAS should be tested for the same items as standalone devices, with these caveats:

1) The Part 15 aspects have nothing to do with Software EAS. We should be able to assume that if a hardware platform passes Part 15 Class A, then it will do so when running EAS software.

---

[8] OET Laboratory Division Equipment Authorization System (EAS), https://apps.fcc.gov/oetcf/eas/ click Authorization Search, Equipment Class EAD.

[9] Trilithic, *EAS Encoder/Decoder test report,* https://fccid.io/P4V-EASYCAP-1/Test-Report/Part-11-Test-Report-1584905.pdf

2) The Part 11 aspects having to do with construction of the EAS elements and their timing, tone frequencies, basic alert reception and transmission can be tested to the same level that a test lab performs on standalone hardware.
3) Environmental concepts such as humidity, temperatures, input power, and THD and modulation levels are more closely tied to the hardware platform. It makes little sense to try to measure this on a device that accepts streaming data as input and sends AES67 as output – the output quality depends on elements far downstream of the EAS software and the hardware that it runs on.
4) Full EAS beyond the ZCZC frame format presence of tones and timing should be tested, as they are now. As discussed above, some EAS procedures have sometimes been self-tested by the manufacturer and passed to the TCB for review.
5) CAP compliance – with lack of an active conformity assessment process, this can also be self-tested by the manufacturer and passed to the TCB for review. If a CA program is available, it can be used.
6) Cybersecurity – not part of current Part 11 EAS testing.

The current test lab/TCB process should continue to be used.

If the FCC requires cybersecurity testing or a declaration of conformance, those declarations could also be reviewed by the TCB.

Fortunately, the Fifth Report and Order[10] allows for testing that is not conducted by FEMA's CA program, or its successor, the NIMS CAP testing:

> *"For integrated CAP-capable EAS devices that do not already have FCC certification, the grantee must include with the FCC certification application materials […] or (c) for devices tested outside these programs, a copy of the test report showing that the device passed the test elements."*

Certifying CAP compliance by a method like the Conformance Assessment is the best choice - but if such a test isn't available and won't become available in a timely manner, then the applicant should be permitted to provide a rigorous test plan and test results and include those results with other test reports submitted to the TCB.

Sage has developed a protype Software EAS product. We found that the software for a Software EAS implementation is not very different from that developed for our standalone product. The testability of the EAS and CAP components is the same. When implemented on a closed shared platform - meaning one that is produced by, for example, an audio processor company that allows curated third-party software that has been approved to run on that platform and has been tested for compatibility with that platform - security can be maintained as both companies work together to achieve that goal. Sage does not advocate running the EAS package on a third-party platform where the software configuration is not controlled, and where the user can install non-curated software on that platform.

In the case of curated hardware and curated software, testability, security and certification become possible and meaningful. This is the environment we are advocating. This is the environment where the existing Part 2 and Part 11 testing is viable, using the existing models and

[10] FCC-12-7, EB Docket No. 04-296, paragraph 167, https://docs.fcc.gov/public/attachments/FCC-12-7A1.pdf

certification rules. The closer we can keep Software EAS to the current methods, the sooner the industry can receive the benefits of Software EAS.

The original test elements from the CA program are still available[11]. FEMA should be able to provide the most recent version of these test elements. FEMA also runs a test and training facility that is used for origination software testing and could be used to generate test messages. The original CA was run before the modern IPAWS Open existed; more and better testing tools are available in 2026 than in 2011.

The best outcome would be to find that FEMA or its partners are able to provide independent testing in a timely manner. If not, it is possible to test the same elements using available test descriptions, either with a willing accredited test lab, or by in-house testing with review by an accredited test lab or directly with a TCB.

Standalone EAS devices have been upgraded over 15 years to accommodate Part 11 changes without an existing CA program. Rigorous testing of EAS compliance can be performed in a variety of ways. Entrants to the EAS market should be permitted to make use of existing data and past precedents to make their case to a TCB.

If a program like the IPAWS Conformity Acceptance program is not available, and all EAS/CAP devices must pass that test, then no new standalone devices or Software EAS implementations can be certified. This would put the EAS system at risk.

In the case of manufacturers with existing products, another path to software certification may be available. Sage's containerized "Software EAS" package uses the same CAP engine that passed the original CA tests and has been updated to current Part 11 requirements. That software component has been running in the field at thousands of locations. We would submit that, in addition to rigorous documented self-testing as proof of adherence to the ECIG CAP rules in a Part 11 certification filing.

Sage believes that the lack of a formal program such as the IPAWS Conformity Assessment Program should not be reason to delay a change to the rules to permit software EAS, nor should it be a reason to delay certification of EAS software, or of devices hosting EAS Software, or of new standalone devices. Self-testing should be permitted upon submission of a rigorous test plan and test results to a TCB, as has been permitted for EAS devices in the past.

As we discuss elsewhere in this document, EAS Software uses the same techniques as standalone devices and must follow the same protocols and procedures. It should be tested to the same standards as standalone devices, no more and no less.

Once installed, the RWT required for most stations provides an opportunity to check for EAS functionality on a weekly basis. Log review is also required. EAS is not a set and forget proposition, periodic monitoring is already required.

[11] *FEMA (IPAWS) Guide for Independent Testing of Emergency Alert System Equipment*, https://www.fema.gov/sites/default/files/documents/fema_ipaws-guide-independent-testing-emergency-alert-system-equipment.pdf

3.15 Additional testing (107)

Some important elements of the Part 11 rules have not been a part of any formal testing. Recent examples are CAP Priority, mandatory immediate NPT, the origination time minus 15 minutes rule in 11.33(a)(10), and possibly more. These items are spelled out clearly in Part 11. EGIC concentrated on items that were not discussed in Part 11 or had more than one possible interpretation. A reconstituted ECIG, an NAB engineering committee, or other respected group, could review the IPAWS CA tests and make recommendations for additional tests.

While broader testing is desirable, standalone EAS devices have existed for many years without that testing. Software EAS, in the interpretation of EAS and CAP messages, and the software's response and adherence to timing, tones, and procedures, is the same as the software running in its custom box-bound brethren. There are no differences or special challenges in Software EAS implementing the same set of rules. In Sage's case, the CAP interpretation software is the same.

Sage sees a need for testing and certification of Software EAS to make sure that EAS and CAP are interpreted correctly, and that alert handling is the same as in a standalone implementation. There is no need for *extra* testing beyond that required for standalone systems, however.

If the FCC wants to expand certification to include new tests, Sage will have no complaints and would assist in specifying new test criteria. All existing devices should then be required to certify adherence to the new requirements by a Class II Permissive Change filing as was required in the Fifth R&O [12]. Software EAS devices should not need to wait for such testing to be defined and become available before they are certified, but they should also make a Class II Permissive filing when new testing or test criteria become available.

3.16 Hardware Testing (108-109)

For hardware, the current EAS specifications were written long ago and assume that the hardware is running in a worst-case environment. As a result, the cost of hardware for conditions encountered by only some users must be borne by all users. Removing the worst-case restriction allows for the use of Software EAS in equipment that matches the actual environment it will be installed in. This covers more normal office studios where business server standards are adequate. It also covers more challenging environments where conditions exceed the current Part 11 rules, where the ability to install Software EAS on a hardened server is of benefit – especially when that server is already present at a facility.

Software EAS need not be constrained to run on hardened hardware. The user can make the decision as to what level of hardware is required based on their facility and should expect higher costs for harsher environments.

We agree with the FCC that Software EAS should be tested on hardware that is representative of that in which it is intended and marketed to be used. However, we do not agree that the full suite of software tests, such the IPAWS CA or response to EAS messages, needs to be run against every possible hardware platform combination in every environmental scenario, as those software elements are not sensitive to the hardware platform. Audio output testing, for example, should be limited to tone frequencies, modulation levels, and those basic audio functions when the hardware platform utilizes physical audio ins and outs. If AoIP is to be used, sending audio

[12] FCC-12-7, paragraph 165, https://docs.fcc.gov/public/attachments/FCC-12-7A1.pdf

to another certified EAS device, possibly through a simple AoIP to analog converter, then checking the resulting output using a frequency analyzer or oscilloscope is sufficient. Quality of the converted audio is more a function of the audio converter than the Software EAS.

This level of testing can be performed by existing Part 11 test labs with allowances for in-situ testing for distributed systems.

As to security, the past Part 11 rules did not stress cybersecurity. The recent R&O requires better passwords and external firewalls or network segmentation; these will improve the security of Software EAS and standalone EAS in equal measure.

For security on a shared platform – yes, the platform must be configured properly to maintain secure access. However, is there any additional risk solely attributable to Software EAS? If a non-EAS device that has access to the air chain is compromised, false EAS messages can be sent over the air. If a device downstream from an EAS device is compromised, it can keep legitimate EAS messages from airing. All parts of a shared platform need to be protected, and all elements of the air chain need to be protected. Software EAS does not add any unique risks.

Multi-vendor software on a single hardware platform is already prevalent. As one example, consider a windows AoIP driver that acts as a virtual sound card and provides a browser interface for configuration. That driver may feed a broadcast automation system on the same computer. This is a potential attack point, and is addressed by signatures on the driver, firewall protection, and keeping up to date with security on the platform. All these items must be configured and patched – often by the local user. Software EAS is no different.

As stated in our Software EAS discussion at the beginning of this document, Sage is not an advocate of a generic EAS driver available for purchase and installation on just any system, and certainly not for the first foray into software EAS. We favor curated software on curated hardware, meaning that the EAS software has been FCC certified for EAS and CAP compliance, and that the host system's other tasks, if any, work properly alongside the EAS software, do not inhibit proper EAS activity, and meet common security goals.

It must be noted that some pairings of EAS software with other non-EAS tasks running on the same device will not be compatible or will require tuning of priority and dynamic loads. It is likely that success for shared platform EAS will be predicated on common testing between, for example, an automation systems manufacturer and the EAS software component manufacturer. Such "acceptable for the purpose intended" testing will be a partnership between those manufacturers to properly service their joint customers.

The software in "Software EAS" and "Standalone EAS" is very similar, and in the case where only AoIP is used for monitor inputs and encoder outputs, it could be identical.

3.17 Recertification (110)

Software EAS is no more likely to need software updates than is a standalone product. Periodic recertification, if required, should apply equally to standalone and Software EAS.

The FCC wants timely security updates. It also issues periodic rules changes that require software updates. A full recertification for each update would delay that process and add to the costs incurred by manufacturers to prepare test reports and pay test labs and the TCB. These would have to be passed on to the end users. If periodic recertification is required, it should be very infrequent.

3.18 Cybersecurity (111-112)

The FCC quotes a commenter to NAB petition, "[u]nlike physical devices, software platforms have large attack surfaces, including APIs, databases, and remote access points." We disagree. A physical EAS device also has many APIs - remote access points, interfaces to SNMP, remote logging, automation systems, web interfaces, printers (network and USB), email, IPAWS,, crawl generators, HD radio exporters, external streaming radios, one-way satellite alert distribution, and two factor authentication are some examples. All of these are risks and must be managed correctly. A proper Software EAS product, of the first three types enumerated in our introduction, has the same risks, and the same mitigation paths. Software EAS thrown on an unmanaged system is problematic, Software EAS in a managed environment on known hardware is far less so.

3.19 Shifting Responsibility (113)

The FCC questions if, as one commenter to the NAB petition asserts, "that EAS software will shift many of the responsibilities for implementing secure hardware configurations, maintaining firmware integrity, performing vulnerability testing, and ensuring end-to-end system hardening, which standalone EAS equipment manufacturers handle today, onto EAS Participants, 'many of whom may lack the technical expertise or resources to manage them effectively'."

When EAS Software is used in an uncontrolled environment, this is possible. However, when one of the controlled implementation types suggested by us in the introduction is used, the user is not left to fend for themselves. The Software EAS provider - Sage, for example - works with the manufacturer of the otherwise closed and controlled environment, for example a broadcast equipment manufacturer, to jointly handle security and interoperability issues. The user's role is to monitor the operation of the equipment, participate in EAS RWT, RMT, and NPT testing, verify the EAS logs, and keep patches issued by Sage and the hardware manufacturer up to date.

The role played by users of controlled Software EAS matches that of standalone system users.

3.20 Expediting Repairs – (114-116)

We agree with the premise that Software EAS can lead to fewer repairs, more redundancy, and less downtime. However, very few problems with Software EAS at a particular user installation will be due to a software bug, that is, a flaw that needs to be fixed by a software update.

Users already need to apply software updates in a timely manner, especially updates for security issues. This was addressed in the recent R&O.

For day-to-day problems, the usual issues are configuration changes, such as network addressing, AoIP channel changes, interface to automation systems, email addresses, firewall settings, and similar issues that affect all broadcast equipment. These problems are the same with Software EAS and standalone hardware. Problems in reception or transmission of alerts that are due to configuration problems should be addressed as soon as practicable.

Software EAS will have additional fail-over and redundancy. Software EAS running on simpler hardware will be less prone to failure.

Software EAS runs on hardware, and a hardware problem will have the same time issues as for standalone EAS. Once the problem is identified as hardware needing repair, 60 days is the more reasonable number.

Software updates and configuration changes are easy to apply in most cases; Sage always tries to provide self-help documentation, email support, and phone support. However, some stations use external engineering resources for this. It will often take more than 72 hours to schedule an on-site support visit.

For cybersecurity issues that lead to software problems, detection, investigation, mitigation, and correction can take varying amounts of time. The nature of this type of problem will sometimes involve several companies, even in the case of standalone EAS systems. Sage has been involved with several issues through the years where the initial error report was that CAP no longer works. These often turn out to be obscure firewall configuration issues where corporate/university or firewall manufacturer support is required and can take more than 72 hours to resolve.

EAS repairs, even for software issues, should be performed as soon as practical, but keep the current 60-day rules.

3.21 Cost of Updates (130)

There are two costs associated with EAS updates. One is the time needed for each user to install the software, or to make changes to configuration files. The other expense is the cost of the update itself. Some companies offer subscriptions. Some periodically have a large paid release with some free updates after that. Some will charge for some updates and not for others. Calculations such as those in the FNPRM's footnote 404 don't take this aspect of major revisions to the rules into account.

3.22 Effective Date FNPRM para 139

The FCC proposes delaying software EAS until twelve months after publication in the Federal Register. We don't see the need for a fixed delay. Historically, EAS rule changes permit the changes as soon as the rule is published but require compliance after certain date.

EAS was created by FCC 94-288, published in the Federal Register December 28, 1994, with an effective date of January 27, 1995. Equipment could be used as soon as it was certified.

When CAP-encoded EAS was first defined, there was only a "must be deployed" date for users, CAP could be used as soon as manufacturers added it via a software update. Certification happened long after CAP equipment was on the market. Manufacturers were ordered to file a Class II permissive change after CAP was deployed.

Sage has shown in this document that

- Software EAS is no different than standalone EAS when it comes to EAS and CAP protocols and procedures.
- The same testing procedures can be used for Software and standalone EAS.
- The lack of an active Conformity Assessment program should not stop new standalone devices from entering the market, nor should it stop Software EAS from entering the market. There is ample precedent for self-testing and review by test labs and TCB.

Software EAS on a closed platform as defined in our introduction, where security and certification is jointly undertaken by the Software EAS manufacturer and the manufacturer of the closed shared hardware platform, is low risk. Software EAS devices like those should not be

delayed by one year after the rules change. Software EAS on those types of devices should be permitted to enter the market as soon as it has been certified.

4. Rules in appendix B

11.2(e)

The FCC's proposed language includes "EAS functions and alerts produced within cloud-based systems, and cloud-based third-party EAS services, are excluded from this definition."

Cloud-based is too loosely defined. The FCC's concern here, as stated elsewhere in the FNPRM, is third-party hardware not under control of the FCC licensee and not at a studio, transmitter, or site associated with the licensed service area. Cloud can also describe the process of using virtualized server-based resources where resources are shared and redundant processes can be instantiated as needed – but on local hardware via a local lan.

We suggest "EAS functions and alerts produced on third-party computing or storage resources, are excluded from this definition. Local shared hardware as described in 11.52(c) is permitted."

11.32 EAS Encoder.

Sage recommends additional clarity for indicators to show that the indicator is not physically part of the hosting hardware platform, but rather on some associated component, via any necessary signaling mechanism, such as LAN, USB, GPIO, etc.

> (a)(7) Indicator. A means of triggering an aural or visual indication, either on the local hardware or any associated component of an EAS system, that is activated when the Preamble is sent and deactivated at the End of Message code.
>
> (9)(vi) A means of triggering an aural or visual indication, either on the local hardware or on any associated component of an EAS system, that clearly shows that the Attention Signal is activated.

11.33(7) EAS Decoder.

Sage recommends that decoder use the "virtual or physical" language used in revisions to 11.31, and remove reference to external or internal speakers

> (7) Outputs. Decoders shall provide at least one physical or virtual data output port where received valid EAS header codes and received preselected header codes are available, and at least one physical or virtual audio output port that is capable of monitoring each decoder audio input.

11.35

Sage recommends 60 days for all cases of "becomes defective" as discussed in "expediting repairs" section of this document.

11.55

Sage recommends using the term “element” rather than “segment”. Element is the term used in the normative section of the CAP 1.2 specification[13] defining area and polygons.

> (d) An EAS Participant that participates in the State or Local Area EAS, upon receipt of a State or Local Area EAS message that has been formatted in the Common Alerting Protocol and that has an event code and CAP area element (using SAME geocodes or polygon/circle coordinates) indicating that it is a type of message that the EAS Participant normally relays, must do the following:

11.56 Obligation to process CAP-formatted EAS messages

Sage recommends unambiguously defining what “valid signature” means.

> (c) EAS Participants shall configure their systems to reject all CAP-formatted EAS messages that do not include a valid digital signature. A valid digital signature meets the requirements of CAP 1.2[14]
>
> and
>
> a chain of trust is established between the alert originator's signing certificate included in the alert's XML data and the trust anchor in the EAS device, in which none of the certificates are expired,
>
> and
>
> the document's cryptographic signature is correct, that is, the digest in the signature matches the digest computed by the receiver and the signed digest was signed by the certificate included in the signature.

Respectfully Submitted,
/s/
Harold Price
President, Sage Alerting Systems, Inc.

[13] OASIS Open, *Common Alerting Protocol Version 1.2* (July 1, 2010), section 3.2.4 "area" Element and Sub-elements https://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html

[14] *Id.* section 3.3.4.1 Digital Signatures