xmatters
AN EVERBRIDGE COMPANY

Financial Services

Large Financial Services Company Centralizes Operations with xMatters

What is your primary xMatters use case?

When we purchased the product, the main reason was for when an out-of-hours issue occurred. Our command center teams were struggling to find the correct on-call resources, and there was a great disparity between teams. Some teams had just their own Excel spreadsheet with who-is-on-call at the time, and some people used SharePoint. Some people were using AlarmPoint, but unfortunately, the old version was quite clunky. It was not very easy to configure the scheduling.

> xMatters has reduced our mean time to resolution because we don't rely on humans looking at screens, picking up an alert, and then picking up a call to get someone out of bed to fix an issue. Now, it is a completely automated process that could save anywhere from 20 minutes to two hours.

INFRASTRUCTURE ANALYST
FinServ company with 10,000+ employees

The primary reason we initially got the product was to consolidate and have one centralized location where all on-call scheduling would be stored. That way, in the event of an issue, the command center, or any team that needs to contact an on-call resource, would be able to easily go to xMatters, search the group, and be able to contact the correct member of staff.

We also use the tool as a notification product for different use cases. Initially, we only had an integration with our alert aggregator for the alerts coming from servers or applications for which we needed to engage a support member for a fix. We plugged that into xMatters, finding over 20,000 events a day. It allowed users to subscribe and say that if there is an alert from a particular server name and the event or alert summary contains X, Y, or Z, give them a call, or send them a text or an email, so that they are aware. Since then, it has grown a bit.

The resilience team has now begun to use xMatters as their primary communication method. So, in the event of an emergency or a security breach, we use xMatters to contact all members of staff across all locations. We have more than 25 locations with over 100,000 members of staff. It's only used in emergencies. It's few and far between that they have to use the product, but they do

The support is excellent. Customer Support always respond quickly. If you need to call them up and get an immediate answer, you can do that. They are happy to help whenever there is an issue.

INFRASTRUCTURE ANALYST
FinServ company with 10,000+ employees

send out quarterly text or voice calls to confirm that they have the right contact details for each member of staff. So, I get a text that says, "This is a B alert, xMatters message. Please confirm that you've received this message." They can then use that to keep track and make sure that they're able to contact everyone in the worst-case scenario or when they have a big problem.

We've now grown into multiple integrations of different products. We are now mainly looking at ServiceNow integration. We do have an in-house-built integration approach where we just use the APIs to pull data from ServiceNow and push it into xMatters, but we are looking at using the actual plugin. The banking world has pretty strict security regulations. The cloud-to-cloud integration needs to go through multiple different hoops, and at the moment, xMatters uses HTTP as opposed to HTTPS. So, currently, we are pulling the data from ServiceNow and pushing it into xMatters. We're not using the official plugin as such. It's just something that we've done via a script by using the APIs of ServiceNow and xMatters.

In terms of deployment, we're accessing our instance through the cloud-based solution that xMatters provides to us.

How has it helped my organization?

We have used coding to expand the flexibility or functionality of xMatters workflows. We've coded a ServiceNow script to use via the API to pull in data and transform it. We also have some parts of the IBM Netcool integration to make sure that we add relevant data or information that we utilize on the xMatters side. There is also some code to make sure that the workflows adhere to our standard or, at least, adhere to our document for the workflow.

We use xMatters' REST API. It is pretty in-depth. It has been a nice feature to have. It has only been around for a year or two, and without it, it was a real struggle. They made major improvements, and you can now pull almost anything that you need to and create a report yourself via the API. You can also integrate with custom workflows that previously would've been a nightmare and required either getting an xMatters consultant on a call or paying them money. Now, with the API, we can insert any data we want. We can target whatever workflows we've worked on ourselves. Overall, it is a pretty good standard of API. You can also request them to add something. If you want to utilize something or pull some data from the API, they're quite receptive to including it in their development plans.

It has helped us to build workflows that meet our needs. The main integration with IBM Netcool is pretty important because, without it, we would be relying on other notification methods or just an email or an alert from ServiceNow which, in our experience, people tend to ignore. When there is an incident raised, making sure they get a text for specific issues is much better. So, we do rely on the product to a certain extent. I'm sure that there are other products out there that can provide the same solution, but it would take time for us to migrate off. With the flexibility of the workflows, we can build what we want on top of xMatters, which is something that we appreciate. It is the sort of thing that we can't live without anymore.

The workflows help in calling out people in the event of an issue, and they also allow us to notify key stakeholders of issues or security-related issues. From a resilience perspective, the tool allows us to send out mass notifications to huge groups of users at the click of a button, as opposed to previous tools that would have us spread across a day to contact the number of people we needed to.

The time saved in terms of creating the on-call rotations and manually doing that each week is about 20 minutes per week. However, if we didn't have this sort of product available to create an alert when an issue happens in the middle of the night, it would require 24/7 eyes on the glass by our team members. They would have to pick up that alert, go to the schedule, find out who's on call, and manually call them and get them out of bed, which could be an hour or two hours process. Even if they see it right away, it is still going to be quite long-winded, as opposed to this automated solution where teams can say that we know what we're going to get called out for, and we can configure that via a subscription, and then they automatically get a call out within a minute or two. There are big benefits to that. Before, for more business-critical issues, it could have taken hours. We're now making sure that people are getting on the calls as soon as possible, as opposed to relying on humans and manual processes.

In a major-incident environment, when the command center needs to reference an on-call rotation, previously, they could have spent half an hour trying to find the correct schedule and figure out who to call. In the end, they probably have to just call a manager who would then know who's on call and call them themselves, which could be time-consuming processes. Now, if you support an application of a certain priority, you're expected to be in xMatters with a rotation, and the command center can then easily reference that in the event of an issue and get hold of the right person.

What is most valuable?

The ability to have the rotation and then configure notifications that you can directly fire into the group is most valuable. The India shift is from 2:00 AM to 9:00 AM, and then it is the UK shift from 9:00 AM to 5:00 PM UK time, and then there is also a defined US shift and on-call hours. It allows us to make sure that everyone is going to get notified when they need to be about an issue. We can target specific locations or users with notifications. One of the main improvements I've seen since the AlarmPoint days is that the rotations are now much easier to configure and use, as opposed to the old days. So, you can configure a rotation and set recurring shifts so that you don't have to do much manual work. In an ideal world, you don't have to log in every week and configure the rotation of who-is-on-call and whatnot. If I'm on-call every four weeks, you can just set that up, and it should recur for you. It saves a lot of time to do the actual tech work, as opposed to on-call rotations work.

It is integrated with IBM Netcool, which is our alert aggregator, and ServiceNow. We have a Spectrum integration in development, which is a network monitoring tool, but we're reviewing whether that's going to be feasible due to the amount of data that Spectrum wants to send in. We try to control the number of integrations we have. Ideally, we want to send everything through IBM Netcool, and then IBM Netcool passes that to xMatters. That's the behavior we are trying to encourage, but if someone comes to us with a use case, we always do consider it. Overall, in terms of the range of integration possibilities, it is pretty good. When I go and have a look at their integration site to see what they have available, I see more and more out-of-the-box applications for me to try and install and test. They've got all the big ones covered. Back in the old days, it wasn't like that.

It has allowed us to send out communications. With the integration that we've got with ServiceNow at the minute, which we've built ourselves, we pull out major incident data and then forward that to some of the senior-level execs in the company. Most of them are not going to be reading emails as much, but they probably will if they get an xMatters notification through their iPhone app or text. If it informs of a major incident, they will obviously look at that. It allows us to make sure that they are kept in the loop. The feedback on the product has been pretty good so far for the people who do use it.

For how long have I used the solution?

We've been using this solution for a good while. We've had at least four years on the SaaS-based solution, but we have been using AlarmPoint and xMatters for over 10 years. So, we are pretty well-versed with the product and its use case.

How are customer service and support?

Their support is excellent. They always respond quickly. If you need to call them up and get an immediate answer, you can do that. They are happy to help whenever there is an issue. They help when there is an issue operationally with a site, and they also seem to promptly assist you when you have basic questions that you are not clear about or would like their advice on. Everything is good on that front. I would rate them a nine out of ten because some of them aren't as helpful as others, but overall, they are very good.

Which solution did I use previously and why did I switch?

Prior to this, I think that we just used SharePoint and Excel spreadsheets to handle on-call data. We have been using AlarmPoint probably since 2005, which is old-world xMatters. It was the old version of the product, and then in 2018, we moved to xMatters, which is a cloud-based solution.

The primary reason for going with xMatters was that we wanted to be able to automate the call-out process specifically for some of our high-priority systems at the mainframe. It is an expensive process to have an L1 person sit there 24/7, even during the quiet hours, just in case one out of 50 times there is an alert from midnight to 5:00 AM UK time. This allowed us firstly to not have to cover those periods because we could just set up an automated process where, if there was an issue, it just calls out the user anywhere. At the time, it was one of the only products available that allowed us to do that with as much configuration as we needed.

How was the initial setup?

It was like a formal project. One of the xMatters consultants came over from the US and worked on it with us in the UK for over three weeks. It was basically a major migration from the old world AlarmPoint to xMatters. We had to make sure that all on-call scheduling were migrated. We had to make sure that it was there in the user devices, and we also had to configure our Active Directory to set up the groups to manage access to each role and trigger the integration agent for IBM Netcool.

What other advice do I have?

I initially was the infrastructure technician who worked on setting up xMatters, and I am now mainly responsible for any escalations with xMatters in terms of operationally managing the SSO-based access user roles. If we do have an issue, which is few and far between, and if it goes to a major incident, I get involved. I'm one of the main users, and I provide L3 support for the product. So, I'm well-versed, and I understand the product well, but at first, it requires a little bit of work. If a user were to log in and configure the office hours shift, it is not the same as just having an Excel sheet that says that ABC is on call Monday to Friday, then XYZ is on call over the weekend. To get the full benefits out of this product, you need a little bit of understanding of shifts, rotations, etc. For that, you have to go and read through the documentation, but these are the things that we expect the users to do. If you're going to be supervising one of the groups, you have to go and read through some documentation to make sure you understand what you're doing and how to configure it. If you don't want to do that, then drag and drop for shifts is pretty simple. If you want to just go and drag and drop a shift each week, you can do that, and it would be extremely simple. You have the options, but it is just that here we try and encourage its users to do it the proper way.

Its on-call schedules and streamlined escalations haven't helped to reduce Sev-1 incidents in our organization. Sev-1 incidents are always going to happen, but it has probably sped up resolution times or at least sped up the engagement for us to make sure that we've got all the right people on the right call. It has also allowed us to notify users, or at least affected users and executives, in a more prompt and efficient manner.

Instead of the in-house-built integration approach, we are now looking at ServiceNow integration by using the actual plugin. Essentially, what we are doing at the minute is that we have a script that makes API calls to ServiceNow to get all major incidents in the last 24 hours. We pull that data from ServiceNow via our script and convert it into a format that xMatters likes. We have a workflow configured for it, and we use the xMatters API to push or post that data into xMatters. The users can subscribe and say that if there's a major incident for my application, they want to get a notification. The actual plugin that we are looking at implementing and getting security approval for seems very simple. It is basically all GUI-based. It is simple to use. There is some work involved in terms of setting up the users in group sync, but hopefully, we'll be able to move to it because there are many more features that we would be able to utilize.

This case study originated from Peer Spot. You can find the original review here.

xmatters
AN EVERBRIDGE COMPANY

xmatters.com