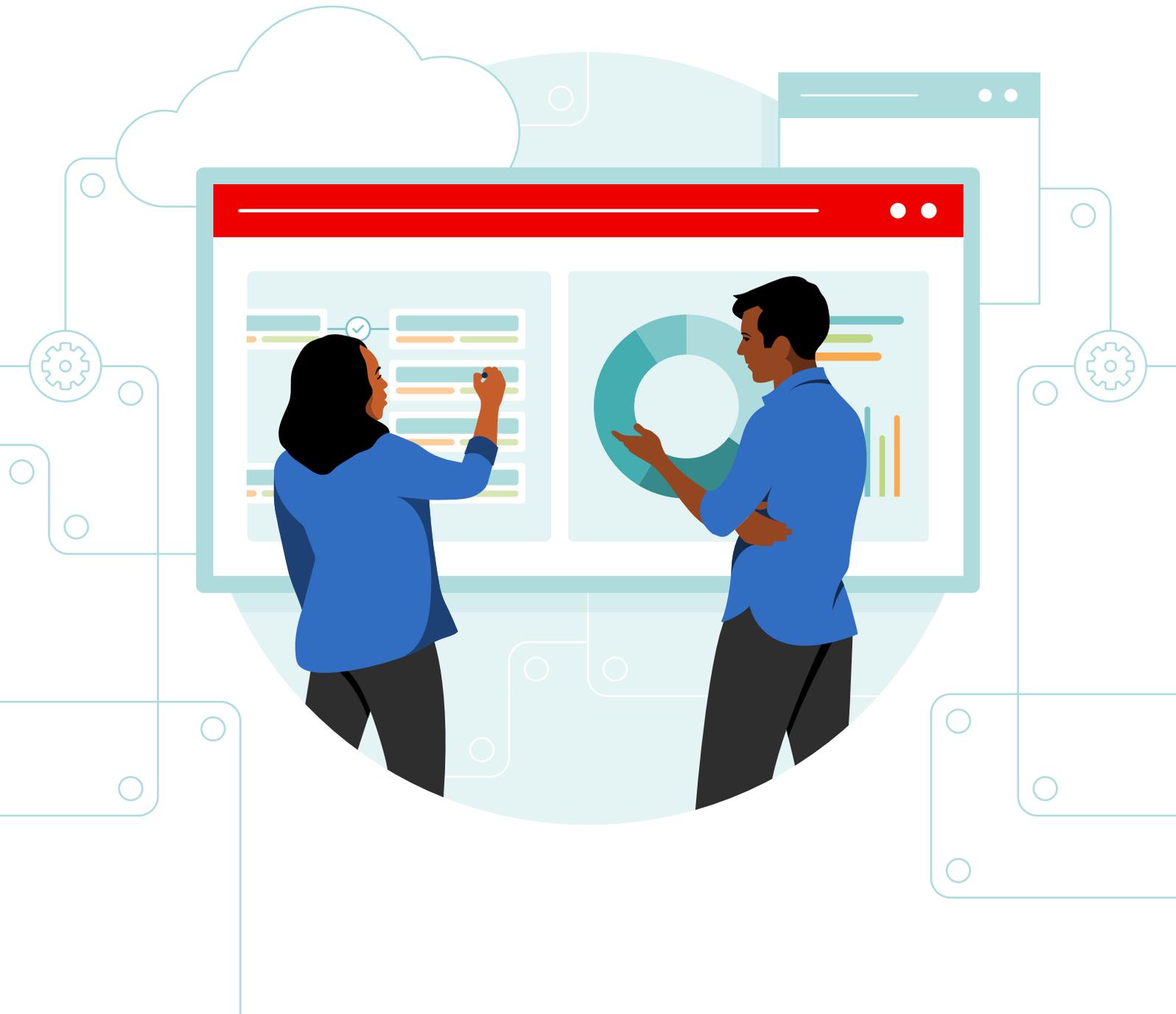


현장의 이야기:
시스템 관리자를 위한
IT 자동화 가이드



3 소개

4 1장: IT 자동화에 관한 일반적인 오해 불식

- 5 내가 IT 자동화를 두려워한 이유
- 8 IT 자동화에 관한 4가지 오해 불식
- 11 일상적인 태스크를 자동화하여 실수를 방지하고 시간을 절약하는 방법

14 2장: IT 자동화의 장점

- 15 수치로 증명: IT 자동화 참여 유도
- 18 자동화를 사용해 증가하는 워크로드에 대처
- 21 운영 제어 확보: 새로운 작업 방식 구축
- 24 고객 사례: IT 자동화로 시간 절약

27 3장: IT 자동화 전문가가 되는 방법

- 28 각광받고 있는 오토메이션 아키텍트
- 32 IT 자동화에 성공하기 위해 필요한 8가지 기술
- 37 Linux 시스템 관리자로서 수입 프로필 및 잠재력을 향상하는 6가지 방법
- 41 자동화 우선 마인드로 전환하기 위한 5가지 방법

44 결론: IT 자동화를 시작하는 방법

- 45 IT 자동화 수용: 커리어에 도움이 되는 이유와 시작하는 방법
- 48 시작하기



소개

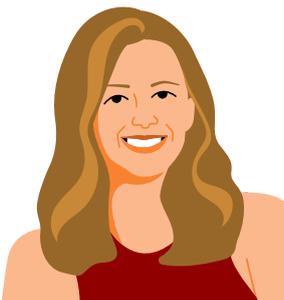
작성자: Emma Van Sant, Red Hat 선임 프로그램 마케팅 매니저

현장의 이야기: 시스템 관리자를 위한 IT 자동화 가이드에 오신 것을 환영합니다. 이 짧은 스토리 모음은 IT 자동화를 전 세계 조직 및 팀에 통합하는 과정에서 겪는 흥미로운 경험과 문제점, 성공 사례, 어려움을 공유하기 위한 것입니다.

IT 자동화만큼 조직의 트랜스포메이션에 영향을 미치는 기술은 거의 없습니다. IT 자동화는 보안 및 컴플라이언스 개선, 비 기술 전문가를 위한 복잡한 태스크 간소화, 조직 전반의 표준화 증진, 조직 확장 지원, 지속적인 서비스 제공 개선, 운영상의 복잡성 및 비용 완화 등의 장점을 제공합니다. 하지만 전체적인 IT 자동화 도입 관점 및 접근 방식을 채택하는 경우 이러한 비즈니스 장점과 맞먹는 어려움이 수반됩니다.

근본적으로 자동화의 관건은 인력입니다. Red Hat® 팀 내에서는 “기술은 쉽지만 사람이 어렵다”라는 말을 자주 합니다. 이 문장이 담고 있는 정서는 신기술 도입과 관련된 기술적 어려움을 호도하는 측면이 분명 있지만 참고할 만한 가치가 있습니다. 최근 연구에 따르면 교육 부족, 인력 대체에 대한 두려움, 조직 문화가 모두 자동화를 광범위하게 도입하는 데 있어서 가장 큰 장애 요소로 나타났습니다. 설문조사에 언급된 10가지 장애 요소 중 6개는 기술이 아닌 사람-문화와 관련된 장애 요소였습니다.¹ 사람이야말로 모든 성공적인 기술 구현 및 도입과 관련해 핵심적인 요소입니다. 승인, 교육, 지원, 계획 수립, 보안 없이는 모든 자동화 도입 이니셔티브가 결국 실패할 수밖에 없습니다.

이 e-book은 바로 이러한 과제에 관한 스토리 모음으로서, 조직 내에서 전체적인 자동화 도입을 가로막는 문화적, 정서적, 현실적 장애로 인해 어려움을 겪은 Red Hat 직원들이 작성한 것입니다. 스토리 작성자들은 조직 내에서 수년 동안 자동화를 구현하거나 수십 년 동안 팀 내에서 자동화 구현 및 도입에 관한 컨설팅을 진행해 왔습니다. 이 e-book은 조직의 IT 자동화 전문가로 자리 잡으면서 IT 자동화가 개인 및 커리어 측면에서 어떤 의미가 있는지 알아보기 위한 좋은 출발점이 될 것입니다.



Emma Van Sant

Emma는 선임 프로그램 마케팅 매니저로서 Red Hat에서 글로벌 IT 자동화 부문을 담당하고 있으며, 지난 7년 동안 체코, 독일, 미국의 IT 조직에서 근무했습니다. Emma는 가장 까다로운 IT 문제를 해결하기 위해 오픈소스 솔루션을 사용함으로써 얻을 수 있는 가치를 전하는 데 열의를 다합니다.

¹ IT 자동화 및 관리 설문조사, 2020년 12월, Qualtrics 및 Red Hat. n=838.

1장

IT 자동화에 관한 일반적인 오해 불식

- 5 내가 IT 자동화를 두려워한 이유
- 8 IT 자동화에 관한 4가지 오해 불식
- 11 일상적인 태스크를 자동화하여 실수를 방지하고 시간을 절약하는 방법



내가 IT 자동화를 두려워한 이유

세 가지 IT 역할의 관점과 이들이 IT 자동화와 관련해 공통적으로 우려하는 사항에 대해 알아보세요.

작성자: Allen Eastwood, Red Hat Consulting 선임 아키텍트

새로운 기술은 많은 약속을 하는 경우가 많지만 경험이 많은 IT 전문가들은 이러한 약속을 경계하듯 대하는 경향이 있습니다. 벤더의 영업팀이 솔루션으로 권유하고 있는 것과 여러분이 해결하려고 하는 실제 문제 또는 과제는 서로 무관하기 쉽습니다. 상당한 변화에 대한 우려가 있을 것으로 예상됩니다. Red Hat의 고객 협력 경험에 따르면 특정 역할은 IT 자동화에 대한 공통된 우려를 보입니다.

경영진

예산 관련 의사 결정을 직접적으로 책임지고 지출 경비와 가장 밀접한 관련이 있을 수 있는 IT 경영진은 두 가지를 가장 우려하는 경향이 있습니다. 첫 번째로, 비용 지불의 대가로 얻을 수 있는 결과를 알고자 합니다. 그리고 두 번째로, 문제가 해결되는 방식을 이해하고자 합니다. 즉, 세부적이고 기술적인 차원이 아니라 프로세스가 개선되거나 문제가 해결되었음을 반영하는 구현 후의 목표가 무엇인지 알고자 합니다.

다행히 IT 자동화의 속성상 이러한 우려 사항은 다른 소프트웨어 솔루션보다 더 해결하기 쉽습니다. 적절한 IT 자동화 프로젝트를 통해 자동화 대상인 시간이 많이 소요되는 수동 프로세스를 정확히 찾아낼 수 있습니다. 태스크와 그 복잡성에 따라 정확한 비용이나 시간을 평가하거나 계산할 수 있는 경우가 많습니다. 더 큰 규모의 프로세스를 뒷받침하는 다양한 태스크를 빠르게 식별하기 위해 노력하거나 “구성 요소” 방식으로 이러한 태스크를 자동화함으로써 진행 과정에서 개선 조치를 점진적으로 결정하고 프로세스가 자동화되고 난 후에 더 큰 규모의 총이익을 파악할 수 있습니다.

IT 매니저

예산 관련 세부 사항에 더 많이 관여하는 IT 경영진과 달리 IT 매니저는 자신이 구매 의사 결정과 관련한 세부 사항에 더 많이 관여하고 있다는 느낌을 자주 받게 됩니다. IT 매니저는 작업 및 워크로드를 관리하지만, 때때로 핵심 인사이트 및 통계가 부족해 직원들이 어떤 방식으로 작업하고 있는지 완전히 이해하지 못하는 경우가 있습니다. 따라서 자동화 활용 사례를 살펴보면 이들은 장기적인 워터폴(waterfall) 방식으로 계획을 수립하는 경향이 있습니다.



하지만 많은 자동화 활용 사례, 특히 컴플라이언스 및 패치 적용과 같이 더욱 복잡한 활용 사례에는 많이 알려지지 않은 종속성이 있습니다. 프로젝트 후반부에 발생하는 이 종속성으로 인해 워터폴(waterfall) 계획 수립의 정확성과 효율성이 저해됩니다. 수동 태스크가 자동화된 프로세스로 점차 교체됨에 따라 일부 알려지지 않은 사항을 밝혀낼 가능성이 있는 프로젝트에 착수하는 경우와 마찬가지로 효율적으로 계획을 수립할 수 없는 경우 충분히 우려할 만한 이유가 됩니다.

과거에 자동화에 초점을 맞춘 컨설팅 프로젝트를 수행하면서 매니저들은 근본 원인과 관련해 지식 격차가 있다는 사실이 공개되어 이를 인정할 수밖에 없다고 느꼈습니다. 하지만 이러한 격차는 이들의 일상적인 작업으로 인해 초래되는 경우가 많았습니다. 많은 매니저들은 비상사태에 끊임없이 대응하고 이런저런 문제에 직원을 투입하는 데 전념하다가 결국 인력이 부족하고 리소스가 불충분하다고 느끼게 되었습니다.

팀, 그리고 팀의 업무 방식에 대해 더 많이 알수록 자동화 프로젝트를 시작할 때 더 적절히 준비를 갖추었다고 느끼게 됩니다.

IT 매니저들은 서비스 티켓을 초래하고 있는 일부 태스크를 자동화하면 결과적으로 대응적인 와해성 관리 업무를 일부 줄일 수 있습니다. IT 매니저들은 팀이 수동 프로세스를 더 빨리 처리하도록 독려하는 대신에 자동화를 사용해 비즈니스 요구 사항을 더 효과적으로 지원하도록 하는 데 집중하게 됩니다.

자동화에 대한 우려를 해소하는 또 다른 방법은 누가 어떤 작업을 수행하고 있으며 어떻게 기여하고 있는지에 관해 전혀 없는 가시성과 통계를 제공하는 Git 및 Jira 툴을 더 깊이 탐구하는 것입니다. 이러한 정보는 앞서 언급한 몇 가지 알려지지 않은 것을 알아내는 데 도움이 될 수 있으며, 자동화 문제를 해결하는 데 시간이 얼마나 소요될지를 더욱 잘 이해하고 예측하는 데도 도움이 될 수 있습니다. 팀, 그리고 팀의 업무 방식에 대해 더 많이 알수록 자동화 프로젝트를 시작할 때 더 적절히 준비를 갖추었다고 느끼게 됩니다.

시스템 관리자 및 엔지니어

위에 언급한 역할들이 자동화에 대해 더 추상적인 두려움을 가졌다면, 시스템 관리자와 엔지니어는 일반적으로 자동화 도입에 관해 훨씬 더 본능적이고 개인적인 두려움을 가집니다. 이들의 일상은 단순한 작업에서 복잡하고 시간이 많이 걸리거나 짧은 작업에 이르는 이러한 수동 태스크 및 프로세스를 중심으로 돌아갑니다. 수동 태스크 및 프로세스를 자동화하면 해당 역할의 담당자를 둘 필요성이 없어진다고 추론할 수 있습니다(이에 대해서는 이 e-book 후반부에서 상세히 살펴봅니다). 이것이 컨설턴트에게 아웃소싱하기 위한 경로일까요? 또한 현 직무에 머문다고 해도 어떻게 유지하시겠습니까? 누구로부터 교육을 받을 것이며, 자동화를 통해 가치를 제공하는 새로운 방법을 어떻게 찾으시겠습니까?

알려진 프로세스가 아무리 지루한 일이라 하더라도 이러한 프로세스에서 벗어나는 것은 실질적인 스트레스 요인이 될 수 있습니다. 이러한 프로세스에서 벗어날 때 본인의 직무가



사라질까봐 걱정하고 있다면 더욱 그렇습니다. 때로는 개인적인 차원의 재구성이 필요하다는 것이 중요합니다. 이전에는 인프라 엔지니어라고 본인을 소개했을 수 있지만 이제 새로운 역할, 즉 인프라 개발자로 발전할 수 있는 기회가 있습니다. 이전에는 수동 태스크를 더 빨리 처리하는 역할을 맡았지만 이제는 자동화를 혁신할 수 있고 비즈니스 가치를 창출하는 데 있어서 더 직접적이고 가시적인 역할을 맡게 되었습니다.

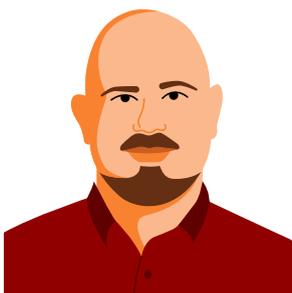
이러한 유형의 전환을 이루는 과정에서 소요되는 노력, 지식, 스트레스를 경시하는 것이 아니라 개발 사례에 대해 새로운 관심을 갖게 된 많은 사람들이 만족할 수 있는 입증된 경로입니다. 그러면 어디서부터 시작해야 할까요? **Red Hat 교육 및 자격증**을 통해 자신 있게 시작할 수 있고, 기술 역량을 높이는 동시에 자동화를 통해 비즈니스 가치를 제공할 수 있는 능력을 상사와 다른 사람들에게 입증할 수 있습니다.

또한 자동화는 유연한 솔루션이므로 새로운 비즈니스 요구 사항에 신속히 적응하는 데 도움이 될 수 있습니다. 동시에 이러한 유연성으로 인해 한 해 동안 어느 위치에서 무엇을 자동화하게 될지 예측하는 것이 어려워질 수 있습니다. 이러한 유연성에 적응하고, 협업적인 자동화 기회를 모색하고, 스택을 이해하려고 노력하고, 플레이북에 대한 거버넌스 시스템을 확립하면 소속된 부서에서 현재 맡은 역할을 보강하고 엔터프라이즈 자동화 달성 경로를 구축하는 데 도움이 됩니다.

결론

자동화에 대한 두려움을 가라앉히는 것은 문화적인 변화를 통해 가능합니다. 개방형 관점은 여러 팀 사이에 존재할 수 있는 일부 세분화된 경쟁 기반 구조에 저항함으로써 자동화 이니셔티브를 가속화하고 가치 실현 시간을 단축하는 데 도움이 될 수 있습니다. 지식 격차에 대한 투명성과 조기 실패를 기꺼이 받아들이는 문화는 궁극적으로 팀이 서비스를 더 빠르게, 더 자신 있게 제공하는 데 도움을 줍니다.

→ **비즈니스 자동화를 위한 5단계**라는 e-book을 읽어보세요.



Allen Eastwood

Allen은 20년 이상의 경력을 보유한 Red Hat Consulting 선임 아키텍트입니다. 고객의 IT 시스템 자동화를 지원하지 않을 때는 아이스하키를 즐깁니다.

IT 자동화에 관한 4가지 오해 불식

IT 자동화에 관한 일반적인 오해를 파악하고 이의를 제기하세요.

작성자: Kedar Vijay Kulkarni, Red Hat OpenShift 네트워킹 부문 선임 소프트웨어 엔지니어

DevOps 엔지니어로 근무하면서 저는 수많은 일상적 태스크를 자동화하는 데 많은 시간을 보냈습니다. 자동화를 이제 막 시작했거나 이미 어느 정도 경험이 있는지 여부에 관계없이 IT 자동화에 관한 여러 가지 일반적인 오해로 인해 저항에 부딪힐 수 있습니다. 제 경험에 근거하여 이러한 오해 중 몇 가지를 반박하고자 합니다.

오해 1: 태스크 자동화는 얻을 수 있는 가치보다 더 많은 시간과 노력이 소요된다.

어떤 태스크를 수동으로 완료하는 데 걸리는 시간보다 자동화하는 데 걸리는 시간이 더 길다면 자동화할 가치가 없다.

시간 절약을 근거로 태스크를 자동화하는 경우 동료나 경영진의 반대에 부딪힐 가능성이 높습니다. 실제로 엔지니어로서 수행하는 모든 작업은 자동화할 만한 가치가 있습니다. 하지만 소요되는 시간과 결과물에 대해 인지하고 있어야 합니다. 어떤 태스크가 자동화할 만한 가치가 없는 것처럼 보이는 경우가 사실은 단지 이번 기회에 자동화할 수 없을 뿐인 경우가 많았습니다. 하지만 향후 목표는 이 태스크를 자동화하는 것이 되어야 합니다. 이러한 관점을 유지하면 팀의 저항이 줄어들 가능성이 높습니다. 즉각적인 목표를 충족하고 향후 효과를 증진하는 방식으로 자동화 제안을 전달하도록 하세요.

오해 2: 일회성 태스크는 자동화할 필요가 없다.

이 태스크는 한 번만 수행하면 되는데 왜 번거롭게 자동화해야 하지?

이것은 제 경력 중에 겪었던 것 중 가장 큰 오해일 것입니다.

실제 예시를 들자면, 한 제품 고객이 특정 문제를 제기했는데 여기에는 특히 복잡한 설정이 필요했습니다. 이 태스크를 수행하던 사람은 제 동료 DevOps 팀원들과 저에게 도움을 청했습니다. 저희는 요청받은 태스크를 자동화하기 위해 최선을 다했습니다. 태스크 소유자가 이를 검증한 후 자동화를 당분간 지속시킬 수 있는지 물었습니다. 그리고 이 “당분간”이라는 시간은 며칠을 지나 몇 주를 넘어 몇 달로 연장되었습니다.

이어서 가상 머신(VM) 인스턴스를 수정해 달라는 또 다른 요청을 받았고, 스냅샷에 이어 결국 VM 복제를 생성해 달라는 후속 요청까지 받기에 이르렀습니다. VM을 자동화로 정리한 후에는 완전히 다시 설정해야 했습니다.



새로운 요청이 들어올 때마다 저희는 요청자에게 해당 태스크를 자동화할 것을 계속 제안했지만 요청자와 경영진은 이를 일회성으로 간주했기 때문에 반대했습니다. 기술적으로는 일회성이었지만 그보다 훨씬 더 많은 노력이 투입되는 작업이었습니다. 태스크를 미리 자동화하는 데 걸리는 충분한 시간을 승인했다면 그들뿐 아니라 저희도 많은 시간을 절약할 수 있었을 것입니다.

그러면 변경이 필요할 때마다 자동화에 대한 업데이트의 일부로 처리하여 모두가 더 편안하게 작업할 수 있었을 것입니다. 이것은 한 가지 예시에 불과합니다. 프로젝트를 시작할 때 문제 해결에 얼마나 많은 작업이 수반되는지 때때로 알기 어려울 수 있지만 한 걸음 물러나 적절한 시간을 두고 너무 늦기 전에 자동화에 투자해야 할지 여부를 평가하는 것이 필수적입니다.

오해 3: 자동화가 중단되므로 자동화하느라 시간을 낭비할 필요가 없다.

자동화가 자주 중단되므로 자동화 유지 관리에 시간을 투자할 만한 가치가 없다.

시간이 지나면서 다양한 스크립팅 언어가 변경되거나 자동화를 통해 상호 작용하는 시스템이 기능을 폐기 또는 도입함에 따라 자동화가 주기적으로 중단되는 것은 사실입니다.

하지만 그렇다고 해서 자동화 노력의 가치가 없는 것일까요? 반드시 그런 것은 아닙니다. 저는 자동화를 사용해 **Red Hat Virtualization** 환경을 구축할 때 이를 직접 경험한 적이 많습니다. 자동화된 태스크는 처음에 4.2 버전에 맞게 설계되었지만 곧 업그레이드를 통해 4.3 및 4.4 버전에 구축할 계획이었습니다.

자동화 유지 관리를 통해 더 큰 확신을 얻을 수 있습니다.

계속해서 자동화를 조정 및 수정하고 각 버전의 다양한 시나리오와 급격한 변동을 처리했습니다. 그 결과 얻은 것은 **Red Hat Virtualization**의 다양한 조합이었습니다. 이 조합은 각 버전에 대한 적절한 입력이 있으면 완전히 배포 및 구성된 **Red Hat Virtualization** 환경을 생성하는 엔진의 역할을 했습니다.

자동화를 디버깅하고 수정하느라 시간을 소비했을까요? 예. 하지만 저는 **Red Hat Virtualization** 환경을 다시 구축해야 할 때마다 이러한 노력이 그만큼 가치가 있음을 확실히 알게 되었습니다. 버전, 크기, 기타 특성이 각기 다른 6~7개의 환경이 있었는데, 새로운 빌드를 사용할 수 있게 되거나 테스트 중 남용으로 인해 환경이 오래된 경우에는 단 한 번의 버튼 클릭으로 리소스를 다시 프로비저닝할 수 있었습니다.

자동화 유지 관리를 통해 더 큰 확신을 얻을 수 있습니다. 자동화를 유지 관리하는 데 시간이 소요되긴 하지만 경험상 처음에 생각했던 것보다 더 효과적이었습니다. 또한 모든 것을 고려했을 때 유지 관리가 포함되지 않는 IT 작업이 있을까요?



오해 4: 이 태스크는 자동화가 불가능하다.

이 태스크는 자동화하기 어렵다. 자동화가 불가능하다.

기대했던 것보다 자동화하기 더 어려운 특정 프로세스를 마주하는 경우가 있습니다. 동료에게서 (또는 인터넷에서 본 정보를 통해) 특정 태스크는 자동화하기 너무 어려워 아무도 성공하지 못했고, 아무도 이를 자동화한 사람이 없으며 불가능할 것이라는 소식을 접하는 것은 드문 일이 아닙니다.

저는 경력 초기에 다양한 인프라 태스크를 자동화하면서 이러한 상황을 겪었습니다. 당시 저는 경험이 부족했고 사람들은 자동화가 불가능하다고 말했습니다. 하지만 당시 제 상사는 제가 이 과제를 계속 추진하기를 바랐습니다. 이 문제의 까다로움을 인정하고 문제 해결에 더 많은 시간을 쓸 수 있도록 기꺼이 배려해준 상사 덕분에 저는 더 크게 동기 부여를 받았습니다.

약 한 달 동안 다양한 옵션을 탐색하고 기술 검증(PoC)을 생성할 수 있는 다양한 방법을 시도한 끝에 해당 상황의 일정한 제약 조건 하에서 태스크를 “반자동화”할 수 있음을 알게 되었습니다. 즉, 수동 개입이 일부 포함되었습니다. 따라서 정말 힘든 일인 것은 맞지만 단순히 불가능하다는 생각은 오해였습니다. 까다로운 자동화 문제를 해결하려면 결단과 용기가 필요합니다.

또 하나의 관점을 말씀드리자면, 한 번도 성공한 적이 없다면 새로운 것을 창안하면 된다는 것입니다. 이것이 바로 제가 한 일입니다. 경험이 더 많은 사람들의 '자동화할 수 없다'는 의견은 옳았습니다. 하지만 그들이 말하고자 한 것은 '완전 자동화는 불가능하다'는 것이었습니다. 엔지니어가 창의적인 문제 해결자의 역할을 하지 못한다면 무슨 소용이 있을까요?

결론

사람들은 자동화에 관한 오해를 그대로 믿는 경향이 있습니다. 자동화에 많은 시간이 소요될 수 있지만 여러분과 팀에게 가치 있는 일이 될 수 있음을 이해하는 것이 중요합니다. 자동화가 중단되고 수정 작업을 해야 하는 경우가 있지만 시간이 절약되고 자동화가 제공하는 장점을 강화할 수 있으므로 그 자체로 가치가 있습니다.

태스크를 자동화하기 어려울 때도 있지만 이러한 어려움을 혁신의 기회로 받아들이고 이를 통해 획득한 지식을 공유할 수 있습니다. 끝으로, 어떤 태스크를 한 번 수행하고 다시 해야 할 가능성이 높을 때 이를 자동화하면 성공의 발판을 갖추게 되는 것입니다.



Kedar Vijay Kulkarni

Kedar는 Red Hat의 선임 소프트웨어 엔지니어로서, Red Hat OpenShift® 네트워킹 작업을 수행합니다. 그의 주무 분야는 소프트웨어 정의 네트워킹의 기능, 성능 및 확장입니다. 이전에는 DevOps 자동화 엔지니어로서 주로 내부 인프라 배포 및 관리에 초점을 맞춰 Red Hat Ansible® Automation Platform, Red Hat Satellite, Red Hat CloudForms®(업스트림 ManageIQ) 프로젝트를 폭넓게 수행했습니다.

일상적인 태스크를 자동화하여 실수를 방지하고 시간을 절약하는 방법

작성자: Shashank Nandishwar Hegde(Red Hat 솔루션 엔지니어) 및 Kedar Vijay Kulkarni(Red Hat OpenShift 네트워킹 부문 선임 소프트웨어 엔지니어)

엔지니어, 시스템 관리자 및 DevOps 역할의 직무에는 수십 통의 이메일에 대한 답변, 수백 개의 서버에 대한 업그레이드, 데이터센터와 선호하는 클라우드 내 모든 시스템에 대한 패치 적용이 포함됩니다. 이 모든 태스크는 완료하는 데 상당히 많은 시간이 소요되며 때때로 지루하지만 항상 중요합니다. 약간의 실수로도 전체 시스템의 보안이 잠재적인 위협에 노출될 수 있습니다. 예를 들어, 보안 패치 적용에 실패하면 시스템이 취약해질 수 있습니다. 솔직히 사람은 실수하게 마련입니다. 하지만 이러한 일상적 태스크를 자동화하여 시간을 절약하고 팀의 태스크를 간소화할 수 있다면 어떨까요?

어떤 태스크를 자동화해야 할까요?

반복 가능한 일상적 태스크

이러한 태스크는 규칙적으로 수행해야 합니다. 예를 들어, 업스트림 리포지토리의 최신 변경 사항을 포함해 내 GitHub 포크를 업데이트해야 하는 태스크가 있습니다. 다른 예시를 들면 다음과 같습니다.

- 시스템 백업 로그 수집
- 매주 시스템 업그레이드 이메일 전송
- 업그레이드 수행

간단하지만 시간이 많이 소요되는 태스크

랩에서 시스템 활용도를 수집하거나 시스템 가용성에 관한 이메일을 매일 팀에 전송해야 할 수 있습니다. 지속적 통합/지속적 제공(CI/CD) 파이프라인 또는 특정 설치와 같이 시스템이 완료하는 데 많은 시간이 소요되는 태스크가 있습니다. 이러한 경우 언제나 몇 가지 경고를 설정하는 것이 좋습니다. 저희는 각 Jenkins 파이프라인 완료 후 수신하는 이메일 경고를 사용합니다. 이와 같은 방식으로 진행 상황을 추적하고, 다른 태스크를 수행하는 데 시간을 할애할 수도 있습니다.

복잡한 태스크

운영 체제(OS) 설치, 최신 패키지 설치, 주별 CI/CD 파이프라인이 시스템으로 실행되는지 확인 등의 몇 가지 작업을 조합해도 복잡하지 않습니다. 하지만 데이터센터나 클라우드 환경에서 수백 개의 시스템에 대해 이러한 작업 단계를 반복한다고 생각해 보세요. 다른 예시로 로드 밸런서 및 프록시 서버 설정이나 도메인 이름 시스템(DNS) 인프라의 시스템에 대한 입력을 들 수 있습니다.



가상 머신 자동화

태스크 자동화의 요점 중 하나는 일상적이고 반복 가능한 태스크를 스크립트화하고 배포 가능하게 하는 것입니다. 이렇게 하면 팀을 위해 더 가치 있거나 전략적이고 사전 예방적인 태스크를 수행할 시간을 확보할 수 있습니다.

자동화할 수 있는 일상적 태스크의 한 가지 예는 Red Hat Enterprise Linux®의 신규 빌드에서 가상 머신(VM) 템플릿을 생성하는 작업입니다. Red Hat Enterprise Linux 7 기반 VM 템플릿을 사용하고 있었는데 Red Hat Enterprise Linux 8이 출시되었다고 합니다. 따라서 모든 템플릿을 새 버전으로 업데이트해야 하는데 이 태스크는 수작업으로 하기에는 너무 힘들고 오류가 발생하기 쉽습니다.

또 다른 일상적 태스크는 팀원이 요청한 VM을 프로비저닝하는 작업입니다. 이러한 태스크를 자동화하면 효율성이 훨씬 높아집니다. 팀이 새로운 VM을 요청할 때 사용할 수 있는 셀프 서비스 포털을 생성하면 훨씬 더 좋습니다. VM에 고유한 구성이 있는 경우 이 요청을 가로채 적절한 조치를 취할 수 있습니다. 또는 자동화가 VM 생성을 처리하고 VM이 준비되었을 때 알림(이메일 또는 채팅 메시지)을 전송하게 할 수 있습니다.

사람들이 잊어버리는 또 다른 중요한 측면은 리소스를 보존하기 위해 VM을 제거하는 것입니다. 엔지니어들은 적절한 시점에 리소스를 정리해야 한다는 것을 잊는 경우가 많습니다. 관리자는 누군가 아직도 VM을 사용하고 있는지 묻은 후에 VM을 수동으로 계속 삭제하거나 VM의 수명에 대한 일련의 룰을 생성할 수 있습니다. VM이 이러한 룰을 준수하는 데 실패하면 VM은 자동으로 삭제됩니다.

자동화를 통해 작업에서 벗어나는 것이 결코 아니라, 어떤 작업을 충분히 수행한 후에 작업의 초점을 바꾸고 싶은 시기를 결정하는 것이 중요합니다.

결론

시간이 많이 소요되는 경우가 많은 단순하고 반복 가능한 태스크의 다양한 예시를 살펴보았습니다. 적절한 계획을 수립하고 몇 가지 툴을 활용하면 이러한 태스크를 완전히 자동화하고 GitHub과 같은 중앙집중식 리포지토리에 저장할 수 있습니다. 이러한 작업 단계를 자동화함으로써 많은 시간을 절약하고 팀의 생산성을 높일 수 있습니다. 가장 중요한 것은 복잡한 태스크가 포함된 경우 실수를 방지할 수 있다는 점입니다. 일반적으로 이러한 태스크는 시스템의 한 세그먼트에서 입력을 받아 다른 세그먼트로 전달한 후 완료 단계로 이동하는 과정에 의존합니다.



항상 모든 작업을 자동화해야 하는 것은 아닙니다. 자동화가 어렵게 느껴진다면 리스크가 적은 소규모 태스크 자동화부터 시작한 다음 다른 태스크를 부분적으로 자동화함으로써 리포지토리를 구축할 수 있습니다. 자동화는 장기적으로 팀의 업무 방식에 큰 변화를 가져옵니다.

요약하자면 자동화를 통해 작업에서 벗어나는 것이 결코 아니라, 어떤 작업을 충분히 수행한 후에 작업의 초점을 바꾸고 싶은 시기를 결정하는 것이 중요합니다.



Shashank Nandishwar Hegde

Shashank는 Red Hat의 솔루션 엔지니어입니다. Red Hat OpenShift 및 Red Hat Ansible Automation Platform 관련 작업에 주력하며 오픈소스 소프트웨어 및 클라우드, 보안, 네트워킹 기술에 열정을 갖고 있습니다.



Kedar Vijay Kulkarni

Kedar는 Red Hat 선임 소프트웨어 엔지니어로서, Red Hat OpenShift 네트워킹 관련 업무를 담당하며 소프트웨어 정의 네트워킹의 기능, 성능, 스케일링에 주력합니다. 이전에는 DevOps 자동화 엔지니어로서 주로 내부 인프라 배포 및 관리에 초점을 맞춰 Red Hat Ansible Automation Platform, Red Hat Satellite, Red Hat CloudForms(업스트림 ManageIQ) 프로젝트를 폭넓게 수행했습니다.

2장

IT 자동화의 장점

- 15 수치로 증명: IT 자동화 참여 유도
- 18 자동화를 사용해 증가하는 워크로드에 대처
- 21 운영 제어 확보: 새로운 작업 방식 구축
- 24 고객 사례: IT 자동화로 시간 절약



수치로 증명: IT 자동화 참여 유도

반복적인 태스크를 자동화하여 더 흥미로운 프로젝트에 집중하는 방법에 대해 알아보세요.

작성자: Andrius Benokraitis(Red Hat Ansible Automation Platform 기술 마케팅 부문 선임 관리자) 및 Cindy Russell(Red Hat Ansible Automation Platform 제품 마케팅 매니저)

열정적인 자동화 지지자의 등장은 성공적인 자동화 프로젝트를 통해 팀이 실현한 결과에서 비롯되는 경우가 많습니다. 수백 개 이상의 시스템, 네트워크, 클라우드 구현 환경 전반에서 업데이트를 일관되고 정확하고 신속하게 완료할 수 있을 때 더 잘 수 있다는 점은 아마도 많은 시스템 관리자에게 가장 설득력 있게 다가오는 혜택일 것입니다. 해결할 문제가 있음을 알리는 한밤중의 호출을 즐기는 사람은 아무도 없습니다.

또한 우리는 보안 및 컴플라이언스 요구 사항과 관련해 팀이 위험을 신속히 해결해야 한다는 압박을 받는 모습을 목도해 왔습니다. 자동화 플레이북에서 적절한 픽스를 생성할 수 있을 때 전 세계 인프라에 있는 영향을 받는 모든 솔루션 전반에서 실행할 수 있습니다. 이 솔루션은 빠르고 효과적이므로 위험을 신속히 해소할 수 있으며, 동일한 작업을 반복적으로 수행할 때 이따금 발생할 수 있는 인적 오류가 없습니다.

하지만 가치를 높이거나 경험 및 기술을 발전시키는 능력을 강화하는 핵심 우선 과제에 더 많은 시간을 할애할 수 있는 등 다른 장점도 있습니다. 팀원 중 일부를 IT 자동화에 참여하도록 유도할 때 제시한 통계 수치는 다음과 같습니다.

몇 시간에서 몇 분 단위로 시간을 단축한 사례

자동화의 장점은 큰 효과로 나타날 수 있습니다. 적합한 자동화를 사용해 “몇 시간에서 몇 분으로” 단축한 사례가 수도 없이 많습니다. 두 가지만 살펴보겠습니다.

20시간 > 30분

Blue Cross and Blue Shield of North Carolina는 단일 가상 머신(VM) 또는 서버를 프로비저닝하는 작업에 20시간 이상(근무일 수 10일 만에 완료)이 소요되었을 뿐 아니라 관리형 서비스 제공업체가 부과하는 요금도 발생했습니다. 현재는 단일 서버에 대한 VM 프로비저닝을 Red Hat Ansible Automation Platform을 이용해 30분 만에 완료할 수 있습니다. 따라서 1,000개의 VM을 신속히 프로비저닝해야 했을 때는 이 프로세스를 완료하는 데 3일이 채 걸리지 않았습니다. 이에 반해 동일한 작업을 비자동화된 방식으로 처리할 경우 12명의 직원이 약 16주 동안 작업해야 완료할 수 있습니다.²

2 Red Hat 고객 사례, “Red Hat과 함께 70,000시간의 근무 시간을 절약한 보험 제공업체.” 2021년 5월.



20분 > 1분 미만

덴버시의 경우 Ansible Automation Platform을 사용하지 않고 개별 Microsoft Teams 환경을 생성하려면 20분이 소요되었습니다. 하지만 Ansible Automation Platform을 사용해 이와 동일한 프로비저닝 프로세스를 진행하자 1분도 걸리지 않았습니다. 팬데믹 상황에서 Microsoft Teams 사용이 514% 증가했지만 자동화를 갖추고 있었기 때문에 엄청난 시간을 절약할 수 있었습니다.³

IT가 이와 같은 요구 사항에 신속히 대응해야 하는 경우 자동화를 활용하면 작업을 빠르게 완료하고 업무 효율성을 높일 수 있습니다. **고객 사례 페이지**에서 이와 같은 더 많은 사례를 찾아볼 수 있습니다.

수요가 많은 자동화 기술

누구나 항상 기술 역량을 향상하고자 애쓰며 이는 모든 기술 전문가의 기본적인 요구 사항입니다. 자동화는 여러분이 갖춘 기술 목록에 추가해야 할 중요한 이머징 기술입니다. 다음과 같은 몇 가지 동향에 대해 살펴보겠습니다.

Ansible은 인기 있는 기술 목록에서 9위를 차지하고 있습니다.

자동화 기술은 설문조사와 동향 리포트에 꾸준히 등장합니다. 2020년에 실시한 **Stack Overflow 개발자 설문조사**에 따르면 Ansible은 가장 인기 있는 툴 목록에서 9위에 올랐습니다.⁴ 그 전년도에 발표된 Indeed.com의 **Hiring Lab 리포트**에서는 Ansible이 2014년에서 2019년 사이에 거의 1,300% 성장하여 세 번째로 빠르게 성장하는 기술로 나타났습니다.⁵

운영에 미친 팬데믹의 영향으로 인해 자동화는 엔터프라이즈 기술 전략의 최전선에 있으며 원격 관리에 도움이 되었습니다.

40,000건 이상의 LinkedIn 채용 공고에서 Ansible 기술 언급

운영에 미친 팬데믹의 영향으로 인해 자동화는 엔터프라이즈 기술 전략의 최전선에 있으며 원격 관리에 도움이 되었습니다. 기업은 자동화의 장점을 직접 경험했고, 이로 인해 엔터프라이즈 IT 자동화 및 관리 전략이 가속화되었습니다. 인터넷을 검색하면 자동화 성장 추세에 대한 많은 애널리스트들의 관점을 확인할 수 있습니다. 이러한 성장 추세는 채용 공고에서도 명백히 드러납니다. LinkedIn에서 검색하면 직무 설명에 Ansible이 포함된 채용 공고가 40,000개 정도 나옵니다.⁶ 확실히 Ansible 자동화 기술에 대한 수요가 많고, 따라서 획득하면 좋은 기술임이 분명합니다.

중요 프로젝트 이행

기업에 최상의 지원을 제공하려면 다양한 기능을 담당하는 IT 팀들이 협업하여 중요한 이니셔티브를 이행해야 한다고 생각합니다. 이러한 이니셔티브의 예로 디지털 트랜스포메이션 애플리케이션, 확장된 하이브리드 클라우드 플랫폼, 현대화된 기술 스택, 엣지 또는

3 Red Hat 고객 사례, "Red Hat Ansible Automation Platform으로 원격 작업을 지원하는 덴버시." 2021년 4월.

4 Stack Overflow, "2020 개발자 설문조사." 2021년 7월 액세스.

5 Flowers, Andrew, "Indeed Tech Skills Explorer: 가장 빠르게 성장하는 기술(Fastest-Rising Tech Skills)," Indeed.com Hiring Lab: 2019년 11월 26일.

6 2021년 6월 23일 기준, 직무 설명에 Ansible이 포함된 LinkedIn 채용 공고 검색에서 40,600건의 결과가 산출되었습니다. 같은 날 "Ansible 엔지니어"에 대한 검색 결과는 훨씬 더 많았습니다.



사물인터넷(IoT) 구현 등이 있습니다. 이러한 이니셔티브는 이머징 기술 학습 또는 사용을 포함하는 경우가 많고, 고객과 기업 리더와 같은 핵심 이해관계자에게 가치를 증명하고 제공하는 데 도움이 됩니다. 이러한 고급 프로젝트에 집중할 수 있는 시간을 더 많이 확보하면 전문가가 발전하고 성장하는 결과를 얻을 수 있습니다.

자동화는 반복적인 태스크를 자동화하고 가치가 낮은 작업을 완료할 수 있도록 지원하므로 더 흥미로운 프로젝트에 집중할 수 있습니다. 예를 들어, 보안 취약점 문제를 해결하기 위해 500개의 시스템에 패치를 적용해야 한다고 합시다. 위 고객 사례에서 보듯이 자동화를 활용하면 필요한 시간이 획기적으로 줄어듭니다. 그 결과 여러분과 기업 모두에 이익이 되는 더욱 혁신적인 프로젝트에 집중할 수 있는 시간을 훨씬 더 많이 확보하게 됩니다.

수동 태스크 부담을 해소하고 다수의 자동화 기술을 보유하게 되면 오토메이션 아키텍트로서 커리어를 발전시키는 데 관심이 생길 수 있습니다. 이 핸드북에서는 “자동화 우선” 커뮤니티를 형성하고 기업의 자동화 전략을 주도하기 위한 모범 사례에 대해 자세히 알아볼 수 있습니다.

여러분과 기업을 위한 “수치로 증명”되는 자동화의 장점을 확인할 수 있기를 희망합니다. 여러분의 학습 여정에 행운을 기원하며 [AnsibleFest](#)에서 자동화에 대한 열정적인 지지자로 만나 뵈 수 있기를 바랍니다.

권장되는 학습 리소스:

- [Red Hat 교육 모듈](#) (무료)
- [온디맨드 및 예정된 웨비나](#)
- [Ansible Automation Platform 동영상](#)
- 무료 체험판: [첫 자동화 프로젝트 구축 및 실행하기](#)



Andrius Benokraitis

Andrius는 Red Hat Ansible Automation Platform 기술 마케팅 부문 선임 관리자입니다. 20년 넘게 컴퓨터 소프트웨어 산업에 종사하면서 IBM, Nortel, Cumulus Networks 등의 기업에서 근무한 Andrius는 네트워크 자동화, 엔터프라이즈 Linux, 비즈니스 분석, 기술 문서 작성, 전략적 제휴 분야에서 전문성을 보유하고 있습니다.



Cindy Russell

Cindy는 Red Hat Ansible Automation Platform 제품 마케팅 매니저입니다. 자동화, 분석 툴, 데이터베이스 및 관련 머신 러닝 기술, 개발자 툴을 포함한 소프트웨어 제품을 위한 기술 실무자를 육성하고 기타 마케팅 프로그램을 개발한 다년간의 경험을 보유하고 있습니다. Cindy는 IBM, Sun Microsystems와 몇 군데의 스타트업 기업, 그리고 소규모의 독립적인 소프트웨어 벤더 기업 등에서 근무했습니다.

자동화를 사용해 증가하는 워크로드에 대처

지루하고 따분한 일상적 태스크에 진절머리가 나십니까? 자동화를 통해 효율성을 높이고 시간을 확보하여 새롭고 혁신적인 기회에 집중할 수 있습니다.

작성자: Ricardo Gerardi, Red Hat 선임 컨설턴트

많은 기업에서 정보 기술 풋프린트가 증가함에 따라 시스템 관리자의 역할도 커지고 있습니다. 현대적인 시스템 관리자는 물리 머신과 운영 체제뿐 아니라 무수히 많은 가상 시스템, 클라우드 환경, 네트워크 기기, 컨테이너 워크로드도 다룹니다. 중소기업에서도 관리되는 시스템의 복잡성과 수는 몇 년 전보다 일반적으로 한 자릿수 이상 증가했습니다.

시스템 관리자로서 여러분의 역할은 배포, 유지 관리, 업데이트, 그리고 이 모든 이기종 시스템이 올바르게 작동하는지 확인하고 고객과 기업이 이러한 기술의 혜택을 누리도록 보장하는 업무로 구성됩니다. 까다로우면서도 흥미진진한 태스크를 수행할 수 있는 풍부한 기회를 제공하는 다이나믹한 직무임이 틀림없습니다. 이러한 직무에는 새로운 기술을 배우고 프로젝트 시뮬레이션 작업을 수행하는 것도 포함됩니다.

하지만 직무의 특성상 그리고 여러분이 관리하는 시스템의 수로 인해 몇 가지 일상적인 태스크도 수행해야 합니다. 이러한 예시로 서버 그룹에 마이너 업데이트를 적용하거나 일부 로그를 제거하는 작업을 들 수 있습니다. 업그레이드에 대비하거나 보안 감사에 대응하는 차원에서 설치된 소프트웨어 버전에 관해 보고하라는 요청을 받을 수 있습니다. 이러한 태스크는 단순하지만 수많은 다양한 기기에서 수행해야 하는 경우 완료하는 데 오랜 시간이 걸립니다. 이러한 태스크에 시간을 소비하면 더 가치 있거나 더 흥미진진한 활동에 투자할 시간이 줄어들습니다.

반복적이고 지루한 태스크를 수행하면서 성취감이 떨어지거나 일에 치인다고 느낀 적이 있으신가요? 그렇다면 혼자가 아닙니다. 수년 동안 동료들에게서 이런 이야기를 들었고 저도 그런 경험이 있습니다.

변화 일으키기

그렇다면 이러한 상황을 개선하고 시간을 더 효율적으로 활용하려면 어떻게 해야 할까요? 이에 대한 답은 복잡할 수 있고, 기업이 이 문제를 체계적인 방식으로 해결하도록 요구하는 경우도 있습니다. 기타 문제의 경우 여러분의 기술 전문성을 활용해 이러한 반복적이고 일상적인 태스크를 자동화함으로써 작업 방식을 개선할 수 있습니다.



IT 프로세스 자동화를 고려할 때는 규모가 크거나 복잡한 문제를 해결하기 위해 자동화 아티팩트를 개발하는 데 드는 시간과 노력을 평가하는 것이 일반적입니다. 여기에는 많은 장점이 있습니다. 단순하고 반복적인 태스크에 자동화를 적용함으로써 이와 동일한 장점을 실현할 수 있습니다. 특히 수백 개 또는 수천 개의 시스템을 관리하는 데 자동화를 재사용할 수 있는 경우 더욱 유용합니다.

이러한 장점의 예를 몇 가지 들면 다음과 같습니다.

- 1. 효율성:** 다수의 특정 대상 전반에서 태스크를 더 효율적으로 실행하므로 컴퓨터가 힘든 일을 대신할 수 있습니다.
- 2. 표준화:** 태스크를 일관된 방식으로 실행하여 매번 동일한 결과를 보장합니다. 예를 들어, 한 대의 기기를 소문자로 구성하고 다른 기기는 하나의 대문자로 구성하는 경우 흔히 헛갈려서 범하는 실수가 나중에 해결하기 어려운 문제로 발전하게 되는데 이러한 문제를 방지할 수 있습니다.
- 3. 오류 감소:** 시스템을 작업하는 사람이 적을수록 중단으로 이어지는 구성 실수 또는 사고가 발생할 확률이 낮아집니다. 예를 들어, DELETE 또는 UPDATE SQL 쿼리에서 WHERE 절을 누락하는 등의 실수로 인한 오류를 줄일 수 있습니다.
- 4. 개인적인 성취감:** 더욱 성취감 있고 지적으로 도전 의식을 불러일으키는 태스크를 수행하면 일반적으로 더 높은 직무 만족도로 귀결됩니다.

어떻게 시작해야 할까요?

자동화 개념을 더 단순한 태스크에 적용하는 것도 자동화를 시작하는 좋은 방법일 수 있습니다. 빠른 성공을 달성하여 즉각적인 혜택을 누리면서 나중에 더 복잡한 문제를 자동화하기 위해 학습하고 준비할 수 있습니다. 사실 저는 몇 년 전에 이러한 방법으로 Ansible Automation Platform을 시작했습니다. 제가 관리하던 몇백 대의 서버에서 주기적인 점검을 실행하는 일부 스크립트를 Ansible Automation Platform을 사용해 교체했습니다. 몇 시간을 들여 초기 구성을 수행했고, 이후 동일한 환경을 사용해 이러한 점검과 기타 많은 소규모 태스크를 자동화함으로써 작업을 빠르고 일관성 있게 수행할 수 있었습니다.

이러한 전략은 자동화 여정을 시작하는 데 효과적이라고 생각합니다. 규칙적으로 수행하고 시간이 많이 소요되거나 번거로운 문제 또는 태스크를 찾아보고 자동화 솔루션을 선택한 후 이를 적용하여 문제를 해결하는 한편, 해당 솔루션과 자동화 전체의 내부 작동 원리를 학습하는 방법입니다.

자동화 개념을 더 단순한 태스크에 적용하는 것도 자동화를 시작하는 좋은 방법일 수 있습니다.



어디서부터 시작해야 할지 모르는 분들을 위해 첫 자동화 프로젝트로 추천할 만한 태스크에 대한 아이디어를 몇 가지 제시하고자 합니다.

- 연결성 테스트
- 패키징 유지 관리: 패키지 설치/업데이트/삭제
- 설치된 애플리케이션 버전 보고
- 사용자 관리
- 사용자 비밀번호 재설정
- 핵심 파일 및 로그와 같은 미사용 파일 제거
- 시스템이나 구성의 백업 또는 복원 실행

다음 단계

단순하고 반복적인 태스크를 자동화하면 태스크를 더 효율적으로 완료하여 더 중요하거나 더 흥미로울 수 있는 프로젝트에 더 많은 시간을 할애할 수 있습니다.

자동화 여정에 어려움이 있을 수 있지만 강력하게 권장해 드립니다. 시도해보면 회사뿐 아니라 개인 차원에서도 많은 장점을 누릴 수 있습니다.



Ricardo Gerardi

Ricardo Gerardi는 Red Hat Canada의 선임 컨설턴트이며, 전문 분야는 Red Hat Ansible Automation Platform과 Red Hat OpenShift를 이용한 IT 자동화입니다. 그는 TELUS에서 선임 아키텍트로 일하면서 통신 부문에서 경력을 쌓았으며, 그전에는 IBM Brazil 및 IBM Canada에서 네트워크 관리 솔루션 담당 선임 컨설턴트 및 사전 영업 스페셜리스트로 근무한 바 있습니다.

운영 제어 확보: 새로운 작업 방식 구축

일상적인 태스크를 자동화 및 간소화하여 제공 시간을 단축하고, 인적 오류를 줄이고, 새로운 작업에 더 많은 시간을 할애할 수 있는 툴을 개발한 고객 사례

작성자: Thomas Tuffin, Red Hat 기술 계정 관리자

여러분은 scrum, kanban 또는 scrumban 기반 프레임워크에 따라 애자일 방법론을 사용하는 운영팀의 시스템 관리자이며, 여러분의 팀에서는 끊임없이 진화하는 환경을 관리합니다. 매일 발생하는 지원 티켓 외에도 마감 시한 안에 이해관계자에게 제공해야 하는 결과물도 있습니다. 이 워크로드에는 많은 노력이 필요하며, 백로그는 갈수록 더 길어지기만 합니다. 이러한 상황에서는 압박감이 큰 환경이 조성되어 이해관계자에게 즉각적인 가치를 제공하는 태스크가 가장 우선시되고 가장 많은 관심을 받게 됩니다. 즉각적인 가치를 제공하지 않는 태스크에는 더 낮은 우선순위가 부여됩니다. 안타깝게도 자동화 툴 및 프레임워크 개발이 낮은 우선순위로 분류되어 팀이 계속해서 수작업으로 일을 처리해야 하는 상황이 너무 흔하게 발생합니다. 이러한 상황이 낫설지 않다면 여러분은 혼자가 아닙니다.

자동화 개발로 관심을 집중시키는 것이 언제나 간단한 일은 아닙니다. 잘 되면 팀의 강력한 지지와 경영진의 긍정적인 지원을 받을 수 있지만 우선 순위를 결정할 때 정당한 사유로 거센 저항에 부딪힐 수 있습니다. 팀에는 서비스 제공 대상인 이해관계자가 있게 마련이며 여러분의 주요 의무는 그들이 필요로 하는 것을 제때 얻을 수 있게 하는 것입니다.

여러분은 운영팀의 일원으로서 다른 팀 및 공급업체와 종속적인 관계를 맺고 있을 수 있습니다. 이러한 관계망에서 한 차례 지연이 발생하면 도미노 효과로 인해 서비스 제공이 지연됩니다. scrum 작업을 하는 팀의 경우 스프린트 종료 시까지 완료하지 못해 다음 스프린트로 넘어가는 사례와 하위 태스크가 드물지 않게 발생합니다. 이러한 백로그로 인해 팀이 받는 압박이 가중되면서 향후 프로젝트가 지연되고 환경 개선에 주로 초점을 맞춘 태스크가 뒤로 미뤄집니다. 이러한 상황에 처한 팀은 마치 통제력을 상실한 느낌을 받게 되고 장기적인 솔루션이 관심을 거의 받지 못하면서 좌절감을 느낄 수 있습니다. 다시 한번 말하지만 이러한 상황은 운영팀에서 흔하게 발생합니다.

자동화를 시작하는 방법

그러면 이러한 상황에 어떻게 대처해야 할까요? 물론 팀, 제품 소유자 또는 프로젝트 매니저, 그리고 필요하다면 상사에게 문제를 제기하는 것으로 시작해야 합니다. 자동화 개발에 리소스를 투자함으로써 절약할 수 있는 시간과 노력에 대한 예시를 제공하는 것도 경영진의 관심을 유도할 수 있는 좋은 방법입니다. 팀이 현재 태스크를 수행하는 데 걸리는 시간, 이 기능을 자동화하는 데 걸릴 것으로 예상되는 시간, 자동화 후 태스크 수행에 걸리는 시간을 간략히 기술한 제안서를 작성하는 것으로 시작할 수 있습니다.



또한 자동화할 수 있는 태스크에 추가 시간을 할당하기 시작할 수도 있습니다. 구성 관리 데이터베이스(CMDB)에 항목을 추가하는 프로세스를 간소화하는 것과 같은 더 작은 규모의 작업으로 시작하세요. 양질의 CMDB에는 애플리케이션 프로그래밍 인터페이스(API)가 있으므로 이러한 API를 이용하기 위해 셸 또는 Python 스크립트를 만드는 작업은 서버 배포 및 관리를 위한 자동화 워크플로우에 필수불가결한 부분입니다. API를 파악하고 이에 접속하는 스크립트가 마련되었으면 이를 한 단계 발전시켜 다른 스크립트가 사용할 수 있는 라이브러리를 구축할 수 있습니다. 라이브러리는 팀이 API와 상호 작용하는 방식을 표준화합니다. 이로써 향후 모든 스크립트는 API에 액세스해야 할 때마다 이 라이브러리를 사용할 수 있으므로 개발 노력이 중복되지 않습니다. 선행 작업을 약간 더 수행할 수 있지만 이로 인해 향후 스크립트 및 툴 개발 시 시간과 노력이 절감됩니다.

CMDB API에 대한 표준화된 액세스 권한이 있는 라이브러리와 자산 정보를 가져올 수 있는 스크립트를 갖추고 나면 서버 배포 프로세스의 여러 부분을 자동화하는 데 착수할 수 있습니다. 배포 절차의 일환으로 PXE(Preboot Execution Environment)와 kickstart 파일을 이미 사용하고 있다면 대부분의 작업을 자동화하지 않을 이유가 있을까요? **Ansible Playbook**을 사용하면 선택한 언어로 작성된 스크립트를 통해 이를 실현할 수 있습니다. 템플릿 모듈과 같은 Ansible의 빌트인 모듈 몇 가지를 활용하면 CMDB에서 가져온 데이터로 PXE 부팅 파일과 kickstart 파일을 손쉽게 생성할 수 있습니다. Ansible에는 다양한 제조업체의 대역 외 관리 인터페이스에 연결하는 데 사용할 수 있는 모듈도 있으므로 대역 외 호스트 이름, IP 주소, 전원 관리 설정과 같은 태스크를 간편하게 수행할 수 있습니다.

물론 다른 툴, 사용자 정의 스크립트, 다양한 오픈소스 및 독점 소프트웨어 솔루션을 통해 동일한 결과를 실현할 수 있습니다. 이 과제에는 여러 가지 방식으로 접근할 수 있으며, 이는 서버 배포에 대해서도 마찬가지입니다. 자유롭게 이용할 수 있는 기술과 툴을 사용해 프로세스를 간소화하고 반복적인 태스크에 소요되는 시간을 줄이세요.

**자유롭게 이용할 수 있는
기술과 툴을 사용해 프로세스를
간소화하고 반복적인 태스크에
소요되는 시간을 줄이세요.**

몇 가지 소규모 태스크를 자동화하고 나면 긍정적인 효과가 나타나기 시작합니다. 먼저, 투박한 CMDB 인터페이스를 클릭하며 정보를 입력하고 검색할 필요가 없습니다. 또한, 서버 배포 프로세스 중 일부가 자동화됩니다. 끝으로, 인적 오류로 인한 문제가 줄어든다는 자동화의 또 다른 큰 장점이 있습니다. IT 산업에서 인적 오류가 다양한 시스템 장애, 중단, 보안 침해의 원인이 되는 경우가 빈번하다는 것은 잘 알려진 사실입니다.⁷ 가장 단순한 태스크, 특히 일상적인 태스크를 자동화하기만 해도 이러한 문제를 줄일 수 있습니다.



환경에 대한 제어 권한 되찾기

더 많은 팀원이 자동화 툴 개발에 참여하면서 툴의 에코시스템이 유기적으로 성장하게 됩니다. 이러한 성장에 따라 코드 품질, 유지 관리 용이성, 책임과 같은 다른 과제가 대두됩니다. 특정 기준을 충족하고 작업 방식을 준수하도록 보장하기 위해 지침을 조기에 확립하는 것이 중요합니다. Git 및 Gerrit과 같은 버전 제어 시스템 및 툴을 사용해 변경 사항을 추적하고 팀 내 코드 협업을 지원함으로써 이를 일부 실현할 수 있습니다.

반복적인 태스크를 처리하기 위해 상태가 양호한 툴셋을 결합한 경우 팀이 자동화를 기본적으로 사용하게 됩니다. 새로운 어떤 것을 계획할 때 팀원들이 가장 먼저 묻는 질문 중 하나는 자동화 가능성 여부일 것입니다. 자동화를 더 많이 구현할수록 현 상태를 변경하고 환경에 대한 제어 권한을 되찾을 가능성이 높습니다. 일상적인 태스크를 자동화하고 간소화하는 툴을 개발하면 이해관계자에게 서비스를 제공하는 시간이 단축되고 인적 오류가 감소할 뿐 아니라 여러분과 팀이 환경의 미래에 대비하는 데 집중할 수 있는 시간을 더 많이 확보할 수 있습니다.



Thomas Tuffin

Thomas는 Red Hat의 기술 계정 관리자입니다. 오스트리아 국적의 스웨덴 거주자인 그는 오픈소스 소프트웨어에 열중하며 블록체인과 같은 이머징 기술에 깊은 관심을 갖고 있습니다. 항상 새로운 것 또는 새로운 장소를 발견하는 데 관심이 있는 Thomas는 여행을 떠나거나 프로젝트에 몰두하는 것을 좋아합니다.

고객 사례: IT 자동화로 시간 절약

새롭고 흥미로운 일에 착수할 수 있는 시간을 확보하고 며칠이 걸리던 태스크를 자동화를 통해 몇 분 만에 완료할 수 있는 방법을 알아보세요.

작성자: Kedar Vijay Kulkarni, Red Hat OpenShift 네트워킹 부문 선임 소프트웨어 엔지니어

2020년에 저는 최신 **Red Hat Satellite** 빌드를 위한 새로운 VM 이미지 생성 프로세스를 자동화하는 팀에서 근무했습니다. 팀의 목표는 VM 배포, 스냅샷, 클린업(cleanup), 템플릿 생성을 자동화하는 것이었습니다. 쉬운 것처럼 보이지만 많은 노력이 필요한 일이었습니다. 자동화는 팀의 시간 절약을 위해 분명히 필요한 일이었으므로 Red Hat Ansible Automation Platform을 자동화 인터페이스로 선택했습니다. 여기서부터 이야기는 시작됩니다.

Red Hat Ansible Automation Platform에서 작업을 한 경험이 있다면 이 플랫폼을 최대한 활용하기 위해 많은 것이 필요하다는 점을 잘 아실 것입니다. 예를 들어, 로그인 및 인증을 설정한 다음, 프로젝트, 자격 증명, 인벤토리, 인벤토리 소스, 작업 및 워크플로우 템플릿, 알림, 일정 등을 설정해야 합니다. 이 모든 작업 때문에 저는 Red Hat Ansible Automation Platform **코드형 구성 ATCasC(Ansible-Tower-Configuration-as-Code)**을 생성하도록 지원했습니다.

이 자동화 프로젝트는 Ansible Automation Platform 사용자 인터페이스에서 수행해야 하는 모든 작업을 YAML 직렬화 언어로 변환합니다. 이러한 설정은 전체 Ansible Automation Platform을 신규 설치에서 완전히 기능적인 서비스로 전환하는 단일 플레이북 명령으로 실행됩니다.

이는 엄청난 장점입니다. 그 이유는 무엇일까요? 구성이 작성되고 나면 코드형 구성(CasC) 방법을 사용해 새로운 인스턴스를 구축하는 데 30분 미만의 시간이 소요됩니다. 이 접근 방식을 사용하기 전에는 새로운 인스턴스를 배포, 설정, 구성하여 프로덕션 레디 상태로 만드는 데 하루 또는 그 이상의 시간(이 작업을 요청받은 사람과 그 사람의 전문성 수준에 따라 달라짐)이 소요되었습니다.

제가 코드형 구성(CasC) 방법을 개발하기 전에는 수동 배포에 1-3시간이 소요되었고 구성은 그 날의 나머지 근무 시간을 모두 할애해야 마칠 수 있었습니다. 프로세스를 가속화하려면 두 사람의 노력이 필요할 것입니다. 예를 들어, 자격 증명 5개, 인벤토리 2개, 인벤토리 소스 2개, 작업 템플릿 20-40개, 워크플로우 5-10개가 포함된 프로젝트가 하나 있다면 마우스 기반 사용자 인터페이스를 통해 이러한 요소들을 생성하는 데 몇 시간이 소요될 수 있습니다. 이 작업을 한 차례 힘들게 완료했다고 합시다. 만일 인스턴스가 손실되면 어떻게 될까요? 어떤 구성도 작성하지 않은 경우 인스턴스를 다시 생성하려면 순전히 여러분의 기억이나 팀 문서에 의존해야 합니다.



바로 이 때문에 반드시 구성을 먼저 작성해야 한다는 점을 알게 되었습니다. 구성을 올바르게 작성하는 것은 팀이 학습할 수 있는 기회가 되었습니다. 그 이유는 무엇일까요? 구성에는 표준 프로그래밍 언어가 없었기 때문에 팀은 YAML 구성체의 스키마를 학습해야 했습니다. 이 학습 곡선을 통과한 후에는 팀원들의 효율성이 더욱 향상되었습니다.

설정 시간을 자동화한 후에는 재해가 발생했을 때 적절한 구성 파일로 즉시 백업하고 실행할 수 있다는 확신을 얻게 되었습니다. 하지만 YAML 구성을 모두 완료하고 테스트함으로써 얻을 수 있는 장점은 무엇일까요?

이 문제를 좀 더 넓은 관점에서 보면, Ansible Automation Platform에서 작업 템플릿으로 실행되는 새로운 플레이북을 작성하는 경우 프로젝트의 YAML 파일에 적절한 프로젝트를 추가한 후 해당 파일에서 적절한 자격 증명, 인벤토리, 작업 템플릿을 추가해야 합니다. 여기에는 최소 50행의 코드가 필요합니다. 이 코드를 이해하고 작성하는 데 30분(경험이 있는 경우)에서 3-4시간(처음 해보는 경우)이 소요될 수 있습니다.

경험이 많아질수록 코드 작성 프로세스는 더 빨라집니다. 반복 가능성과 일관성을 확보하게 되므로 시간을 들일 만한 가치가 있습니다. **코드형 인프라(IaC)**의 장점과 단점을 모두 적용할 수 있습니다.

그다음에는 구성(코드)를 테스트할 수 있습니다. 바로 이 단계에서 저희 팀은 또다시 몇 시간을 들여 프로덕션 인스턴스처럼 보이고 제안된 모든 변경 사항을 포함하는 테스트 인스턴스를 구축했습니다. 이어서 병합 요청을 완전히 테스트하는 데 필요한 작업이 무엇인지 파악하고 마침내 병합하고자 했습니다. 이 작업은 하루나 이틀이 꼬박 걸리는 워크로드였습니다.

자동화를 이용해 테스트 문제를 해결하기 위해 팀은 GitLab 지속적 통합(CI)을 사용해 자동화된 접근 방식을 고안했습니다. 저희가 구현한 자동화를 통해 새로운 풀 요청(PR)이 개설될 때마다 GitLab CI가 이 PR에 대해 새로운 테스트 인스턴스를 생성합니다. 인스턴스 배포 태스크를 누가 수행하느냐에 따라 자동화를 통해 2-4시간을 절약할 수 있었습니다. 이제 GitLab이 인스턴스를 배포하게 되었으므로 더 많은 시간이 절약되었습니다.

그다음 과제는 풀 요청을 어떤 방식으로 테스트할지 알아내는 것이었습니다. 더 작은 규모의 PR 몇 개만으로 무엇을 테스트할지 빠르고 쉽게 알아낼 수 있었습니다. 복잡한 PR은 십여 개가 넘는 파일을 작업하므로 PR을 마스터 브랜치로 병합하기 전에 적절히 테스트하지 않는 경우 어떤 것이 중단될지 예상하기 힘들었습니다. 프로덕션 인스턴스는 마스터 브랜치의 코드를 기반으로 실행되고 있었다는 점에 유의하시기 바랍니다.

PR을 분석한 후 테스트하는 데 드는 시간을 줄여야 하는 과제를 해결하기 위해 저희는 Ansible Genealogist라고 하는 새로운 프로젝트를 고안했습니다. 이 프로젝트에서는 몇 분 내로 PR을 검토한 후 테스트해야 하는 대상에 대한 내용을 문서로 작성합니다.

경험이 많아질수록 코드 작성 프로세스는 더 빨라집니다.





태스크	수동 작업 시 소요되는 시간	자동화 사용 시 소요되는 시간*
새로운 Ansible Automation Platform 인스턴스 배포 — 프로덕션 레디	~1-2일	< 30-45분
테스트 인스턴스를 프로덕션으로 푸시하기 전에 새로운 구성을 테스트하기 위해 테스트 인스턴스를 배포하고 구성	~4-6시간	< 30-45분
새로운 각 PR에 대해 무엇을 테스트해야 하는지 결정	~1-2시간	< 5-10분
테스트 실행	~2-6시간(복잡한 PR의 경우 더 오래 걸림)	< 5분(자동화 테스트 스크립트를 작동시키고 나중에 다시 돌아와 결과 확인)
일부 중단으로 인해 프로덕션 단계에서 실행 중이던 것이 손실되었을 때 프로덕션 인스턴스를 다시 배포	추산 불가. 재해 상황이므로 문제 해결에 모든 구성원을 동원(팀원들이 수행해야 할 일을 잘 알고 있고 임무를 완수하는 경우 아마도 1-2일 미만 소요)	< 30-45분
프로덕션에 새로운 작업 템플릿 추가 또는 자격 증명 업데이트와 같은 변경 사항 적용	꺼려지는 태스크 — 실수가 나쁜 결과로 직결. 프로덕션을 업데이트하기 전에 변경 사항을 테스트하기로 한 경우 최대 1일의 작업 일수 예상	< 30분 — 변경 사항이 PR 프로세스 진행 중에 이미 테스트되므로 기본적으로 CI/CD 중 CD를 통해 프로덕션으로 푸시

여기서 볼 수 있듯이 며칠이 걸리던 태스크를 자동화를 통해 몇 분 만에 완료했습니다. 자동화할 태스크가 계속 생겼기 때문에 직무 자체를 자동화한 것은 아니었습니다. 저희 그룹의 목표는 가상 머신을 위한 표준 관리 태스크인 배포, 템플릿, 스냅샷 등을 자동화하는 것이었습니다. 시간 절약은 이 프로젝트에서 매우 중요한 부분이었습니다. 또한 재해 발생 시 반복 가능한 이벤트를 생성하기를 원했습니다. 자동화와 템플릿을 통해 재해 복구 상황에서 효율성을 훨씬 더 높일 수 있는 기회를 얻게 되었습니다.



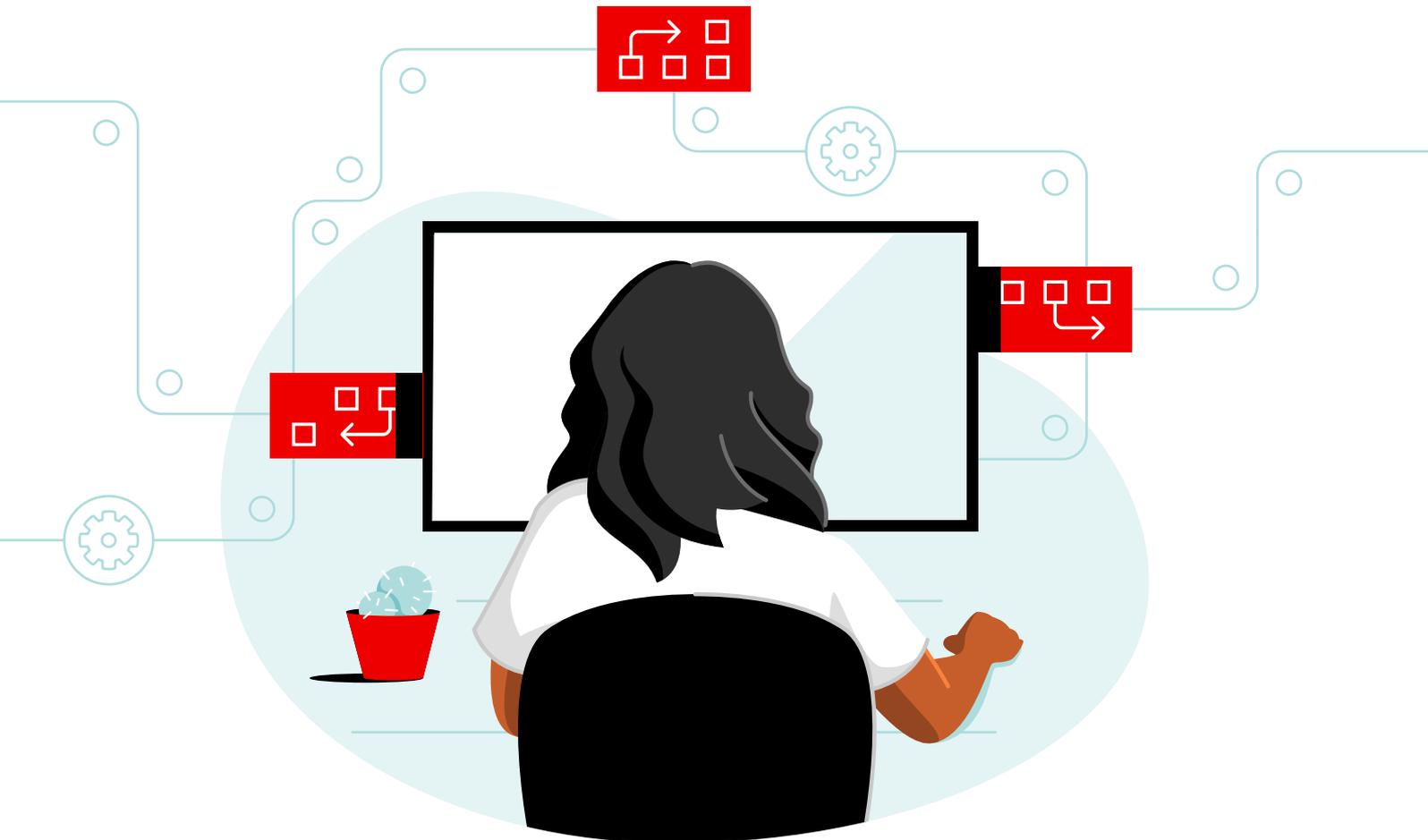
Kedar Vijay Kulkarni

Kedar는 Red Hat 선임 소프트웨어 엔지니어로서, Red Hat OpenShift 네트워킹 관련 업무를 담당하며 소프트웨어 정의 네트워킹의 기능, 성능, 스케일링에 주력합니다. 이전에는 DevOps 자동화 엔지니어로서 주로 내부 인프라 배포 및 관리에 초점을 맞춰 Red Hat Ansible Automation Platform, Red Hat Satellite, Red Hat CloudForms(업스트림 ManageIQ) 프로젝트를 폭넓게 수행했습니다.

3장

IT 자동화 전문가가 되는 방법

- 28 각광받고 있는 오토메이션 아키텍트
- 32 IT 자동화에 성공하기 위해 필요한 8가지 기술
- 37 Linux 시스템 관리자로서 수입 프로필 및 잠재력을 향상하는 6가지 방법
- 41 자동화 우선 마인드로 전환하기 위한 5가지 방법



각광받고 있는 오토메이션 아키텍트

여기에서 제시하는 팁을 사용해 IT 커리어를 발전시키고 오토메이션 아키텍트로서의 입지를 정립하세요.

작성자: Joseph Tejal, Red Hat 기술 계정 관리자

대부분의 사람들은 두려운 미지의 영역이던 자동화 과제를 해결하면서 진화하는 새로운 역할인 오토메이션 아키텍트가 등장하는 것을 바로 눈앞에서 보았습니다.

자동화는 2015년부터 많은 관심이 집중되는 주제였습니다. 이때부터 Ansible과 같은 오픈소스 프로젝트가 부상하면서 널리 사용되기 시작했습니다. 많은 흥미와 관심을 불러일으켰지만 운영 센터의 입장에서는 많은 회의와 두려움이 있는 것도 사실이었습니다. 현재로 빨리 거슬러 올라와 보면 자동화는 현재 대부분 조직의 기술 전략에서 핵심을 차지하고 있습니다. 이러한 전반적 추세는 계속 발전하면서 더 복잡해지고 있습니다.

자동화를 통한 혁신 증진에 관한 [Forrester 설문조사](#)에서는 기업들이 중요성을 다루는 일련의 목표 전반에서 자동화에 더 우선순위를 두고 있음을 보여줍니다.* 기업들은 자동화가 기업의 계획에 대해 갖는 중요성을 인식하고 비즈니스의 모든 측면을 자동화하기 위해 노력하고 있습니다. 이들 기업이 성공을 거두려면 누군가 나서서 이러한 노력을 선도해야 합니다.

이제 다시 돌아가 자동화 아키텍처가 어떻게 오늘날의 모습으로 발전하게 되었는지 살펴보겠습니다.

자동화 여정의 시작

처음에 운영팀과 각 직원들은 자동화가 쉽고 재미있다고 생각했습니다. 대부분 개발 관련 경력이 없었지만 몇 가지 일상적이고 틀에 박힌 태스크를 자동화하는 방법을 배웠습니다. Ansible과 같이 배우기 쉬운 툴 덕분에 매우 열정적인 직원들에게는 자동화가 쉽게 느껴졌습니다. 시스템 관리자에서 인프라 및 데이터베이스 엔지니어, 심지어 서비스 데스크 직원에 이르기까지 모든 사람이 자동화하는 방법을 배웠습니다.

커피를 마시며 가볍게 주고받는 이야기에서 복도를 오가며 나누는 대화, 경영진을 설득하기 위한 실제 데모에 이르기까지 관심이 커지고 있었습니다. 머신 프로비저닝, 애플리케이션 재시작, 패치 적용 및 유지 관리를 수행했고 이 외에도 자동화 목록은 계속 늘어만 갔습니다. 사람들은 협업과 아이디어 제안에 나서고, 자신의 영역에서 실행되는 멋진 결과물과 서로 협력할 수 있는 방법에 대해 이야기하기 시작했습니다.

하지만 자동화의 작은 개별 조각들을 연결하면서 팀들은 문제에 봉착하기 시작했습니다. 그 이유가 뭔지 아마도 아실 것입니다.



문제에 직면

얼마 지나지 않아 상황은 더 복잡해졌습니다. 다루어야 할 활용 사례가 더 많아졌고, 자동화 루틴 및 워크플로우를 서로 연결할 필요가 있었습니다. 저희는 어려움을 겪기 시작했고 어떤 사람들은 경영진으로부터 점점 더 중대한 요구 사항을 받으면서 자신이 거둔 일시적 성공의 희생자가 되기도 했습니다.

자동화를 진행하면서 겪을 수 있는 몇 가지 문제는 다음과 같습니다.

- 중요성을 다투는 우선순위와 목표
- 표준, 정책 및 거버넌스의 부족
- 세분화된 직장 문화로 인한 장애물
- 필수적인 몇 가지 기술의 부재
- 구현 관련 보안에 대한 우려 사항
- 재사용성, 확장성, 제어 능력에 대한 고려 부족

많은 운영팀이 이러한 문제를 오히려 기회로 삼습니다. 이러한 문제들을 처음 겪는 사람들은 혼란을 느끼고 주인의 의식 및 책임성 결여로 인해 방향성을 잃기 시작합니다. 하지만 이러한 문제는 환경과 태스크가 아무리 복잡해져도 성공적인 자동화 사례를 구축할 수 있는 선도적인 위치를 점유할 수 있는 기회를 제공합니다.

성공을 위한 오토메이션 아키텍트의 역할

전체적인 자동화 접근 방식을 실행하려면 기술된 성공을 위한 필수 역량을 탐색할 수 있는 추진자가 필요합니다. 이러한 역량은 인프라 자동화 성숙도에 관한 [Forrester Research](#)에 간략히 설명되어 있습니다.¹⁰

이러한 역량에는 다음이 포함됩니다.

- 전략
- 프로세스
- 우선순위 지정
- 인력
- 구조
- 운영

조직이 지향해야 할 방향성을 알고 목표에 성공적으로 도달하는 방법을 알 수 있도록 여정을 주도할 오토메이션 아키텍트가 필요합니다. 오토메이션 아키텍트의 역할은 위에 언급한 문제를 해결하여 모두가 자동화를 통해 자동화의 가치를 극대화하도록 돕는 것입니다.

[Red Hat의 오토메이션 아키텍트 핸드북](#)에 설명된 것처럼 조직을 효과적으로 운영하는 데 필요한 중요 태스크는 다음과 같습니다.



자동화의 현재 상태를 이해하고 목표를 설정

이 프로세스를 통해 아키텍트는 유용할 수 있는 기존 툴과 역량을 식별하고 이 외에 필요한 것이 무엇인지 파악할 수 있습니다. 지연과 우회하는 상황이 발생하더라도 모두가 순조롭게 올바른 방향으로 나아갈 수 있도록 조정하는 실현 가능한 목표를 설정하고 우선순위를 파악해야 합니다. 이러한 노력을 통해 모두가 목표를 염두에 두고 필수적인 태스크에 우선적으로 집중할 수 있습니다.

결속과 협업 증진

격차가 지속되고, 의견이 서로 다르고, 업무 방식이 상충할 수 있습니다. 오토메이션 아키텍트는 비즈니스 목표 및 우선순위에 노력을 집중하고 이에 연계함으로써 모든 사람을 결속시켜야 합니다. 이러한 노력은 모두가 공개적으로 참조하고 논의할 수 있는 잘 정의된 요구 사항으로 전환되어야 합니다.

팀이 협력하도록 장려하는 개방형 협업 접근 방식 및 툴을 사용하세요. 단편화된 그룹을 해체하고 결속시키는 활동을 포함하여 연대와 포용을 통해 성공을 거두세요.

**오토메이션 아키텍트는
비즈니스 목표 및 우선순위에
노력을 집중하고 이에
연계함으로써 모든 사람을
결속시켜야 합니다.**

자동화 자산 및 역량 향상

자동화 인벤토리를 배치하고 나면 모두가 집단적으로 리소스에 대한 작업을 시작할 수 있도록 리소스를 핵심 리포지토리에 중앙집중화하세요. 보안 및 모범 사례에 초점을 맞춰 표준, 적절한 검토, 버전 제어를 설정하면 여러 팀 전반에 신뢰할 수 있는 방법을 제공할 수 있습니다. 그러면 팀은 팀 구조를 기반으로 효과적인 학습 방식을 사용해 이러한 툴과 워크플로우에 기반을 둔 역량 격차를 해소하기 위한 노력에 착수할 수 있습니다. 다양한 툴에서 학습할 수 있는 무료 리소스가 많이 있습니다.

모든 사람이 자동화 관점 및 문화를 발전시키도록 유도

자동화 옹호자 및 지지자를 늘리는 것은 전체적인 접근 방식을 창안하고 자동화가 어떤 방식으로 작동하고 팀에게 이익이 되는지에 대한 이해를 공유하는 데 있어서 핵심이 되는 요소입니다. 오토메이션 아키텍트는 자동화에 대한 신뢰, 열정, 관심을 증진할 수 있는 창의적인 방법을 모색해야 합니다. 이러한 접근 방식은 간단한 식사와 함께 진행하는 가벼운 회의 세션, 성공 대시보드, 데모에서부터 재미있게 접근할 수 있는 코드 마라톤(code-a-thon) 대회에 이르기까지 다양하게 구현될 수 있습니다.

적절한 거버넌스 및 관리로 자동화 솔루션 확장

양질의 프레임워크가 마련되면 보안 및 모범 사례를 희생하지 않고도 팀 내에서 자동화의 규모와 비중을 늘릴 수 있습니다. 오토메이션 아키텍트는 거버넌스를 확립하여 조직의 정책을 준수하고 제어 및 강력한 감사 기능이 포함된 솔루션을 통해 사람들이 책임 의식을 갖게 해야 합니다. 또한 적절히 관리 및 식별되는 위험 수준 내에서 성공하고 혁신할 수 있는 기회를 모두에게 제공해야 합니다.



확고한 입지를 다진 오토메이션 아키텍트

갈수록 빠르게 진화하는 복잡한 기술의 시대에 오토메이션 아키텍트와 같은 새로운 역할이 각광받고 있는 추세는 고무적인 일입니다. 이는 시스템 관리자, 분야별 전문가(SME), 사이트 신뢰성 엔지니어링(SRE), 엔지니어가 운영 및 자동화 여정 전반에서 자신의 시간을 최대한 효율적으로 사용할 수 있는 훌륭한 기회입니다. 자동화의 힘을 통해 제한된 전문 분야 밖으로 진출함으로써 기술 수준을 높이고 전문성을 넓히고자 하는 의욕을 갖게 됩니다. 이들은 조직의 기존 과제 및 고충을 이해하고 분석할 수 있는 전략적 위치를 점하게 됩니다. 이를 기반으로 성공적인 엔드 투 엔드 자동화를 위해 다른 사람과 협업하고 최상의 접근 방식 및 툴을 사용하는 등 비즈니스 목표에 맞게 솔루션을 조정함으로써 이러한 문제를 어떻게 해결하는지 보여줄 수 있습니다.

이 역할을 다 발전적으로 수행하려면 자동화 관련 문제를 더 잘 이해하고 이러한 문제에 적합한 솔루션으로 접근할 수 있도록 지속적으로 성장하고 학습해야 합니다. **오토메이션 아키텍트 핸드북**에서 자세한 내용을 알아보세요.



Joseph Tejal

Joseph은 Red Hat의 기술 계정 관리자입니다. 본질적으로는 Unix 시스템 관리자로서 서비스 개선 이니셔티브에 참여하고 Ansible을 사용해 일상적인 태스크를 자동화합니다. 다수의 고객과 함께 자동화 프로젝트 제공 관련 작업을 수행했고 뉴질랜드에서 Ansible 및 OpenShift 모임을 활발하게 공동 주관해 왔습니다.

IT 자동화에 성공하기 위해 필요한 8가지 기술

스크립팅, 협업, 소스 코드, 관리 등으로 자동화 기술 향상

작성자: Chad Ferman, Red Hat 선임 애플리케이션 배포 솔루션 아키텍트

지금 되돌아보면 운이 좋았습니다. 저는 2000년이 되기 전에 이미 자동화 팀을 보유하고 있던 IT 회사에서 제 경력을 시작했습니다. 이 팀은 문제가 발생하기 전에 선제적으로 문제를 해결하거나 적어도 문제가 발생했을 때 한밤중에 누군가를 깨우지 않고 문제를 해결하는 마법과도 같은 능력을 보유하고 있었습니다. 당시 19세였던 저는 이것이 표준이라거나 자동화가 진정으로 의도하는 것이 아니라는 생각은 전혀 하지 않았습니다. 그러던 어느 날 문득 든 생각은 이렇게 하면 밀려 있는 다른 업무를 모두 처리할 수 있는 시간 여유가 생기고, 더 중요한 것은 다시는 수작업으로 이러한 업무를 처리할 필요가 없다는 것이었습니다. 커맨드라인에서 수행할 수 있는 모든 것을 손쉽게 코드로 저장하고 인간의 개입 없이 체계적으로 다시 실행할 수 있다는 깨달음은 제 삶을 완전히 변화시켰고 오늘날 제가 이 자리에 있게 된 계기가 되었습니다.

자동화의 힘

기업 전체의 자동화 전략을 책임지는 아키텍트 역할을 수행하던 전 직무에서 제가 되뇌던 신조는 '자동화가 단지 스크립팅에 국한되는 것은 아니다'라는 것이었습니다. 자동화는 이를 훨씬 더 뛰어넘는 것이므로 먼저 이에 대해 간략하게 설명하겠습니다. 스크립트는 자동화의 기반이지만 스크립트가 자동화의 전부는 아닙니다. 무언가를 반복 가능하게 하려면 시스템이 어떤 상태이고 제대로 작동하고 있는지 알려주는 정보가 필요합니다. 바로 이 점에서 관측성과 모니터링이 중요합니다. 이 두 가지 기술을 이용하면 최종 목표 달성을 위해 프로그래밍 방식으로 완료해야 할 작업을 정보에 근거해 결정할 수 있습니다. 수신되는 정보와 외부로 이동하는 자동화로 구성된 피드백 루프가 있다면 서비스 전달이 지속적으로 향상되는 순환 구조가 생성됩니다.

성공적인 커리어 관리를 위해 이러한 기술을 추가해야 하는 이유는 무엇일까요?

자동화는 애플리케이션 개발에서 인프라 배포와 비즈니스 프로세스에 이르는 모든 것을 지원합니다. 가치를 창출할 수 있는 기회는 무한히 제공됩니다. 몇 가지만 예를 들면, 여러분은 DevOps 엔지니어, 사이트 신뢰성 엔지니어, 애자일 코치, 제품 소유자, 통합 엔지니어, 인공지능(AI)/머신 러닝(ML) 운영 또는 비즈니스 프로세스 엔지니어의 역할을 수행할 수 있습니다. 여러 시스템이 비즈니스 가치를 제공하기 위해 서로 어떻게 상호 작용하는지 이해하는 능력은 많은 산업에서 원하는 기술이며, 자율적으로 작동하는 반복 가능 프로세스를 만드는 일을 좋아한다면 이것이 바로 여러분에게 맞는 직무입니다.



더 적은 인력으로 더 많은 일을 하는 것은 많은 조직이 해결하고자 하는 과제로, 특히 경기 침체기에는 더욱 그렇습니다. 수동 태스크를 제거하여 개선 작업에 할애할 수 있는 시간을 확보할 수 있습니다. 이 접근 방식은 신뢰성과 예기치 않은 다운타임의 시기 적절한 해결을 통해 경영진의 신뢰를 높여줍니다. 성공적인 자동화 패턴은 모든 사람이 공동 목표를 이해하고 이 목표를 중심으로 결속하는 데 도움이 됩니다.

예를 들어, 저는 성공적인 자동화 패턴을 입증함으로써 더 큰 책임과 기회를 부여받았습니다. 이전 직장에서 저는 기존 프로세스를 개선하여 3년 만에 DevOps 엔지니어에서 엔터프라이즈 아키텍트로 승진했습니다. 몇 달 걸려 서버를 배포하던 수준에서 30분 만에 풀 스택 애플리케이션 서버를 제공할 수 있게 발전했고, 구성되고 실행되어 비즈니스 가치를 제공하는 비즈니스 애플리케이션의 엔드 투 엔드 배포를 실현했습니다.

자동화는 재미있는 작업이어서 성취감이 굉장히 클 수 있습니다. 40단계로 구성된 CI/CD 파이프라인을 실행하여 보안 및 API를 검증하고, 코드 분석 및 린팅을 수행하고, 사용자 인터페이스 요소가 올바른 위치에 있는지 확인하고, 대시보드에서 파이프라인의 상태를 '녹색/성공'으로 표시하는 회귀 분석 테스트를 수행하는 것을 지켜보는 것은 매우 짜릿한 경험입니다.

자신이 수행하는 일의 가치를 알고 함께 일하는 사람 중 누구나 이 일을 반복 수행할 수 있다는 것을 알면 큰 성취감을 느낄 수 있습니다. 내 변경 사항이 다른 사람에게 영향을 미치지 않으며 다른 사람의 변경 사항이 내 작업을 중단시키지 않는다는 것을 알게 되는 경우에도 역시 마찬가지입니다. TV를 시청하시겠습니까, 아니면 드라마를 직접 연출하시겠습니까? 느긋한 자세로 작업이 의도한 대로 진행되는지 확인하고, 테스트에 실패하는 경우 이에 대한 알림을 받아 나중에 수정할 수 있습니다.

자동화하려면 어떤 기술이 필요할까요?

자동화를 위한 최소 실행 가능 기술(MVS)에 대해 들어보셨나요? 이러한 기술에는 스크립팅, 협업, 소스 코드 관리, 쿠버네티스, 보안, 테스트, 관측성, 모니터링, 네트워크 인식 등이 포함되며 이에 국한되지 않습니다.

스크립팅

플랫폼에 내장된 언어(Windows용 PowerShell 또는 Linux용 Bash)로 능숙하게 스크립팅할 수 있는 능력을 갖추기 위해 노력하는 것이 훌륭한 출발점이 될 수 있습니다. 하지만 자동화 수준이 더욱 복잡해지면 Python과 같은 범용 언어를 이해하는 것이 바람직합니다. Python을 언급하는 이유는 이 언어가 지난 10년 동안 네트워크, 서버, 스토리지, AI/ML을 위한 사실상의 표준으로 자리 잡았기 때문입니다. 자동화 프레임워크 전체가 이 언어로 작성되어 왔습니다.

협업

스크립팅은 이해하겠는데 협업은 왜 필요할까요? 진정한 엔드 투 엔드 자동화를 가능케 하려면 여러 팀이 서로 협력해야 합니다. 극소수의 사람들만이 네트워크, 스토리지, 방화벽, 프록시 등이 실제로 어떻게 작동하는지 알고 있으므로 이 모든 것을 한데 묶으려면 공통성이 있어야 합니다. 바로 이 때문에 자동화 코드를 어떻게, 어디에 저장하고 어떻게 배포할 것인지에 대한 기업 전략을 마련하는 것이 매우 중요합니다. 코드뿐 아니라 아키텍처 설계를 공유할 표준 위치와 아키텍처의 각 부분과 통신할 API가 없다면 고객을 위한 인프라, 애플리케이션, 서비스 제공을 적절히 자동화하기란 불가능합니다.



소스 코드 관리

모든 코드를 GitHub, GitLab, Azure DevOps 또는 Bitbucket과 같은 **Git** 관리 툴로 중앙집중화하면 다른 팀뿐 아니라 그룹 내 사람들과 훨씬 더 쉽게 협업할 수 있습니다. 버그를 찾은 경우 문제를 해결하고 이를 효과적으로 문서화(단순히 “손상됨”이라고 쓰지 않도록)하는 데 익숙해지는 것은 아직 코드 커밋을 시작할 준비가 안 된 경우 훌륭한 출발점이 될 수 있습니다. 문제에 익숙해졌다면 코드에 대한 약간의 풀 요청과 커밋 수정을 시작하고 다른 사람이 입력한 풀 요청을 검토하여 그 기능을 테스트하도록 지원할 수 있습니다. 완벽한 사람은 없으므로 사람들은 코드 검토에 대해 매우 고마워하며 다른 사람이 고려하지 않은 부분을 여러분이 찾아낼 수도 있습니다.

애플리케이션 프로그래밍 인터페이스(API)

모든 사람이 작업할 때 사용하는 API와 플레이북의 중앙화된 목록을 만드는 것은 자동화의 성공을 위해 매우 중요한 일입니다. 이 기술은 API를 단순히 애플리케이션에 포함하는 것에 그치지 않습니다. 서비스에 접속하는 방법을 묻기 위해 여러분을 호출하지 않고도 누구나 사용할 수 있게 하는 것은 서비스 전달을 자동화하는 최상의 방법입니다. 이로써 개발자가 기존 인프라 서비스가 필요할 때 티켓을 입력하거나 전화를 하지 않고도 요청할 수 있는 클라우드 리소스와 똑같아집니다.

컨테이너와 쿠버네티스

컨테이너와 쿠버네티스는 현대적인 애플리케이션을 하이브리드 클라우드 전반에 배포하는 실질적인 방법으로 자리 잡았습니다. 컨테이너를 구축한 후 배포, 확장, 모니터링, 재배포하는 방법에 대한 확실한 이해는 기업들이 강력히 원하는 능력입니다. 이 기술은 머신 러닝 및 애플리케이션 개발에서 비즈니스 인텔리전스 및 사이버 보안에 이르기까지 기업의 다양한 영역에 두루 적용됩니다. 컨테이너를 이용해 로컬 머신에서 실행되는 구성 요소가 모든 플랫폼의 애플리케이션 수준에서 동일하게 작동하도록 보장할 수 있습니다. 쿠버네티스로 컨테이너를 배포하기 위한 다음 단계를 진행할 때 모든 환경 전반에 걸친 원활한 배포를 위해 필요한 모든 것이 마련되도록 코드를 통해 보장할 수 있습니다.

네트워크 인식

네트워크 인식에 관한 대화를 시작할 때 저는 항상 “클라우드가 물리학, 즉 빛의 속도를 해결하지 못한다”라고 말합니다. 데이터와 사용자가 어디에 위치하고 데이터 컴퓨팅이 어디서 진행되는지 이해하는 것이 매우 중요합니다. 첫째, 최종 사용자가 액세스할 데이터와 비교해 최종 사용자의 위치를 이해해야 합니다. 처리 작업이 사용자와 동일한 위치에서 진행되도록 확인해야 합니다. 이에 실패하는 것은 흔한 실수로서 반복적으로 발생하며 애플리케이션 성능을 약화시킵니다. 물론 잘못된 배치가 아닌 애플리케이션이 비난을 받습니다. 대기 시간을 고려하는 것은 특히 대역폭이 양호한 국가에 있는 경우 더욱 당연하게 여기는 또다른 사항입니다. 대기 시간에 대한 왕복 테스트를 실시하여 최종 사용자가 어떤 경험을 할지 반드시 확인하세요. 예를 들어, 모든 최종 사용자가 싱가포르에 있는데도 텍사스 주에서 애플리케이션을 빌드하기를 원하는 경우가 있었습니다. 오랜 숙고 끝에 싱가포르에서 이 애플리케이션의 플랫폼을 재구성했고, 사용자들이 필요로 하는 것을 이 애플리케이션에서 얻을 수 있도록 사용자 환경을 구축하는 데 소요되는 시간을 몇 분에서 몇 초로 단축했습니다.



테스트

테스트는 있으면 좋은 기술 정도로 치부되는 또 하나의 기술이지만, 한 번의 키보드 입력 실수로 프로덕션 환경을 중단시키는 사태를 막아줍니다. 구동시킨 부분이 기대한 대로 작동하는지 검증하는 것은 성공을 거둘 뿐 아니라 문제 해결을 위해 밤새도록 작업해야 하는 의도치 않은 결과가 발생하지 않도록 보장하는 데 있어서도 극히 중요한 작업입니다. 백엔드 테스트로만 국한해서는 안 됩니다. 변경 사항이 기존 기능이나 최종 사용자 환경에 영향을 미치지 않도록 사용자 인터페이스(UI) 요소 및 API를 테스트 및 검증할 수 있는 훌륭한 툴이 있습니다.

보안

애플리케이션에 보안을 구축하는 것은 암호 화폐를 채굴하기 위해 클라우드 배포를 가로채는 악의적인 사용자와 랜섬웨어가 만연한 오늘날의 환경에서 매우 중요한 작업입니다. 보안 통합은 애플리케이션을 배포하는 CI/CD 파이프라인에 포함되어야 합니다. 이 파이프라인 내에는 여러분에게 필요한 핵심 요소들이 있습니다. 즉, 배포 시 생각한 것과 동일한 코드 또는 아티팩트인지 확인하기 위한 정적 코드 분석, 아티팩트 관리 및 추적, 보안 라이브러리, 코드 서명입니다. CI/CD는 애플리케이션 인스턴스화만 처리합니다. 배포하려는 대상 플랫폼에 대한 보안 강화도 필요합니다. 또한 알려진 취약점이 있는 라이브러리를 실행하지 않고 있으며 라이브러리에 패치를 적용해야 하는 경우 팀에 알릴 수 있음을 입증하는 조치가 필요합니다.

사이버 범죄가 증가함에 따라 보안 테스트 업무는 많은 기업에서 채용하고 있는 또 하나의 직무입니다.

사이버 범죄가 증가함에 따라 보안 테스트 업무는 많은 기업에서 채용하고 있는 또 하나의 직무입니다. 애플리케이션 공급망에 보안 테스트 기능을 내장하는 작업은 빌드에서 배포를 거쳐 코드가 의도한 대로만 실행되고 있는지에 대한 런타임 검증에 이르는 라이프사이클 전체에서 많은 기업들이 수행하고 있는 추세입니다. 서명된 라이브러리 및 실행 가능 파일은 컨테이너와 같은 라이브러리 및 아티팩트 소스의 검증과 마찬가지로 표준으로 자리 잡아가고 있습니다. 신뢰할 수 있는 서명된 라이브러리 및 컨테이너를 사용하고 사용자 정의 라이브러리와 아티팩트 리포지토리를 조직에 제공하는 것이 보안을 중요시하는 모든 조직의 표준이 되었습니다.

관측성과 모니터링

애플리케이션의 상태와 이러한 상태를 실현한 방법에 대해 이해하는 것 역시 태스크를 적절히 자동화하는 데 필요한 기술입니다. 서비스와 관련해 어떤 일이 발생하고 있는지 알지 못한다면 사전 예방적인 자동화를 생성하여 문제를 해결하거나 향후 발생할 수 있는 문제를 방지하는 일관성 있는 상태를 적용할 수 없습니다. 대부분의 사람들은 모니터링에서 그치기 때문에 문제가 발생하면 근본 원인 분석(RCA)을 사용해 어떤 일이 발생했는지 알아봅니다. 관측성은 RCA에 필요한 툴을 제공하므로 RCA 기능을 상시 활용해 모니터링을 통해 알게 된 가동 또는 중단 상태 외에도 모든 상황을 파악할 수 있습니다. [DevOps 모니터링 가이드](#)는 이에 관해 자세히 알아볼 수 있는 훌륭한 자료입니다.



요약

저는 자동화야말로 오늘날의 현대적인 IT 세계에서 사람이 수행할 수 있는 가장 보람 있는 직무 중 하나라고 생각합니다. 이 직무를 수행하려면 거시적인 사고방식과 각 요소의 전체 작동 방식에 대한 이해가 필요합니다. 여러분이 문제를 해결하는 사람으로서 다른 사람에게서 “이게 작동 방식입니다”라는 말을 듣고만 있을 수 없다면 이 직무는 여러분을 위한 것입니다. 수작업으로 하는 일을 반복 가능한 프로세스로 만들어 사람들이 더 가치 있는 일에 집중할 수 있게 할 때마다 회사의 비용을 절약(우리는 모두 많은 경우 결국 비용 문제라는 것을 알고 있습니다)할 뿐 아니라 사람들이 훨씬 더 흥미로운 프로젝트를 수행하도록 돕는 것입니다. 위에 나열된 모든 기술은 서로 상승 효과를 내어 여러분이 더 뛰어난 자동화 전문가로 거듭나도록 도움을 줍니다. 이러한 기술은 갑자기 획득할 수 있는 것이 아니므로 여유를 가지고 편히 즐기면서 수작업을 중단하시면 됩니다.



Chad Ferman

Chad는 Red Hat 선임 애플리케이션 배포 솔루션 아키텍트입니다. 그는 공공 및 민간 부문의 유통 및 오일/가스 분야에서 25년 넘게 엔터프라이즈 IT 전문가로 일했으며, 인프라 운영에서 마이크로서비스 애플리케이션 개발과 엔터프라이즈 전략에 이르기까지 모든 것을 두루 다루었습니다. 최근에 Red Hat에 합류하여 고객이 엔터프라이즈 소프트웨어 배포와 현대적인 방법론으로의 문화적 트랜스포메이션을 성공적으로 실현할 수 있도록 지원하고 있습니다.

Linux 시스템 관리자로서 수입 프로필 및 잠재력을 향상하는 6가지 방법

커리어 또는 급여 향상이 필요하다면 적절한 경로로 안내하는 몇 가지 훌륭한 팁을 알려드립니다.

작성자: Joseph Tejal, Red Hat 기술 계정 관리자

소개

상사와 한자리에 앉아 여러분의 성과를 논의하는 시기가 되었을 때 핵심이 되는 문제는 대화를 통해 급여를 높이거나 승진 대상에 포함될 수 있는 기회로 만들 준비가 되어 있는지 여부입니다.

최상의 시나리오는 많은 말을 하지 않아도 되는 경우입니다. 조직에 기여한 성과와 가치는 그 이상의 설명이 필요 없으므로 상사는 자신의 삶을 더 수월하게 만든 여러분께 고마워하며 이를 승진의 명분으로 삼습니다. 이는 조직 내에서 성장함으로써 시스템 관리자로서의 수입 잠재력을 향상할 수 있는 여러 방법 중 한 가지에 불과합니다.

직장 외부에서 다른 기회가 다가오는 경우 면접 중에 잠재적 고용주가 채용을 결정하도록 확신을 심어줄 수 있는 성공 사례가 준비되어 있으신가요? 최상의 시나리오는 여러분의 이니셔티브와 성공 사례를 자연스럽게 공유하고 고용주가 관심 있게 경청하는 과정에서 더 많은 이야기를 듣고 싶어 하고, 결국 좋은 조건으로 여러분을 채용하게 되는 것입니다. 훨씬 더 좋은 것은 여러분이 속한 현지 커뮤니티에서 여러분의 프로필과 브랜드가 매우 우수하여 여러 회사가 여러분을 채용하기 위해 경쟁하는 것입니다.

시스템 관리자로서 수입 잠재력을 향상할 수 있는 몇 가지 방법이 있습니다. 중요한 문제는 '이러한 입지에 어떻게 도달하고 이러한 대화 및 기회를 어떻게 준비해야 하는가?'입니다.

여기서는 제 경험과 제가 대화를 나눴던 다른 사람들의 경험에 근거하여 몇 가지 팁, 조언, 아이디어를 공유하고자 합니다.

자신을 없어도 되는 사람으로 만들기

직관에 반대되는 말로 들리겠지만 저는 발전하기 위해서는 기존 태스크를 뒤로 하고 가치를 높이는 데 집중해야 함을 알게 되었습니다.

팀과 조직을 위한 가치 증진 — 여러분이 수행하는 일상적인 태스크를 누구나 할 수 있도록 **개선, 자동화**, 문서화하세요. 더 좋은 것은 아무도 그 일을 할 필요가 없어지는 것입니다. 주로 겪는 고충을 해결하는 데 집중하여 누구나 여러분의 역할을 자신 있게 수행할 수 있게 하세요. 이를 통해 더 가치 있는 이니셔티브에 참여할 시간을 확보할 수 있고 이러한 참여에 대해 모든 사람의 신뢰를 얻을 수 있습니다.



현 상태를 고수하지 않고 자기 자신과 규준에 도전하세요. 특히 이러한 규준이 효율적이지 못하고 이미 오래된 경우 더욱 중요합니다. 이해관계자는 여러분이 창출하는 가치와 개선하는 사항에 깊은 인상을 받을 것입니다.

커리어와 목표를 주도적으로 관리

커리어와 목표를 관리함으로써 자신과 조직의 한계를 극복할 수 있습니다. 때때로 필요한 지원을 제공해줄 수 없다는 소식을 듣게 됩니다. 하지만 주도적으로 학습하고 제공된 리소스를 활용하여 발전할 수 있습니다. 특정 기술 교육을 요청하기 전에 실험해볼 수 있는 수많은 오픈소스 프로젝트가 있습니다.

온라인으로 제공되는 무료 체험판과 워크숍을 통해 **Red Hat Ansible Automation Platform**, 하이브리드 클라우드, 쿠버네티스, **Red Hat OpenShift**와 같은 새로운 기술에 관한 조사와 기술 검증(PoC) 연구를 수행하세요. 이러한 리소스는 회사의 향후 학습 및 개발 관련 투자에 적합한 후보자가 되도록 지원하므로 공식 교육 및 **자격증 시험**의 자격을 갖추는 데 도움이 됩니다.

누군가에게 책임 의식을 가지면 학습에 매진하기 더 쉬우므로 함께 학습할 동료, 기술 코치 또는 멘토를 찾기를 원할 수도 있습니다. 단지 기술적인 측면의 발전이 아니라 리더십 포지션으로 수직 성장하고 다른 시스템 관리자의 매니저 역할로 경력을 개발하기를 원할 수도 있습니다.

단지 기술적인 측면의 발전이 아니라 리더십 포지션으로 수직 성장하고 다른 시스템 관리자의 매니저 역할로 경력을 개발하기를 원할 수도 있습니다.

소프트 스킬 개발

요즘 시대에는 기술 역량을 갖추는 것이 수입 잠재력을 향상하는 유일한 방법은 아닙니다. 감성 지능, 태도, 타인과 협력할 수 있는 능력, 의사소통 스킬은 여러분만의 고유한 장점과 차별성을 부각시킬 수 있는 몇 가지 요소입니다.

조직 내에서 여러분이 존경하는 동료나 코치를 찾는 것으로 시작해 보세요. **Toastmasters**와 같은 스피치 클럽에 가입하여 회의실과 연설 무대에서의 대인 관계 및 의사소통 스킬을 향상할 수도 있습니다. 글쓰기 역시 주목을 받을 수 있는 한 가지 방법입니다. **Enable Sysadmin**, **opensource.com**, **medium.com**과 같은 다양한 공간에 참여하고 다른 이들로부터 배울 수 있습니다.

익숙한 환경에서 탈피하는 것에 당황하거나 두려워하지 마세요. 저는 키보드, 단말기, 기술 전문성과 관련 없는 영역에서 자신의 능력을 발견하고 스스로 놀란 많은 사람들을 알고 있습니다.



다른 사람들과 협업

일부 시스템 관리자만 일급비밀을 알고 혜택을 누리고 나머지 사람들은 모두 중요 지식에서 소외되던 시대는 지나갔습니다.

지금의 협업을 통해 혁신이 창출되는 시대입니다. 다른 팀들과 협력하여 자신의 지식을 확장하고 다른 팀의 지식 확장에도 기여해야 합니다. 관심이 가는 팀에서 자발적으로 순환 근무를 할 수도 있습니다. 팀 내에서 몇 가지 프로세스를 개선하는 소규모 프로젝트를 진행하거나 현재 보유한 것보다 더 나올 수 있는 업데이트된 툴 및 기술을 탐색해 보세요. 이러한 협업을 통해 부서 간 구분을 해체하고, 문화적 변경을 시작하고, 관심을 불러일으킬 수 있으며, 이를 통해 모든 사람을 성공으로 이끌 수 있습니다.

참여하도록 권장을 받은 사람들이 손쉽게 액세스할 수 있도록 모든 사람의 작업과 진행 상황을 공개적으로 문서화해야 합니다.

다른 사람들과의 소통을 통해 프로필 향상

커리어 상의 소통 관계 및 레퍼런스는 여러분에게 이익이 됩니다. 여러분이 함께 일했던 주요 인물들의 훌륭한 피드백을 통해 여러분의 상사는 여러분이 발휘할 성과와 기대치 능가 여부를 짐작할 수 있습니다.

또한 커뮤니티에 참여하면 미래의 고용주가 될 수도 있는 다양한 사람들과 소통할 수 있습니다. 회사 또는 지역 커뮤니티에서 점심을 곁들인 학습 기회 또는 모임과 같은 **이벤트에 참석하거나 이러한 이벤트를 주관하세요**. 이러한 이벤트에 참석하면 다른 사람들에게 도움이 되고 자신을 알릴 수 있습니다.

팀이나 부서 내에서 소규모 그룹 또는 간단한 식사와 함께 하는 세션을 시작하여 여러분의 성공 사례와 흥미로운 프로젝트에 대해 논의할 수도 있습니다. 익숙해지면 기술 회의 및 참여 행사에서 현재 추진 중인 멋진 프로젝트에 관한 서류나 제안서를 제출함으로써 프로필을 더욱 발전시킬 수 있습니다.

즐겁게 일하기

의욕을 북돋는 일을 찾아보세요. 때로는 일이 벅찰 수 있지만 여러분이 어떤 사람인가는 어려울 때 대처하는 방식에 따라 결정됩니다. 커리어 전체 또는 커리어의 상당 부분을 즐기고 있다면 도움을 주고 가치를 높일 수 있는 이러한 상황에 참여하는 것에 흥미를 느낄 것입니다.

이 상황을 즐기고, 여러분의 스킬과 장점을 보여줄 수 있는 기회를 모색하고, 약점을 개선할 수 있는 기회를 포함하세요. 시야를 넓혀 새로운 열정을 발견할 때 일은 더 이상 지루하지 않습니다.

다른 사람과 관계를 맺고, 말은 영역에서 창의성을 발휘하고, 평범한 일상 밖에서 기회를 찾음으로써 직무 수행에 재미를 느낄 수 있는 방법이 많이 있습니다. 업무 시간 외에 재충전하는 시간을 가짐으로써 활기를 되찾으세요.



협업을 통해 열어 가는 밝은 미래

수입을 얻고 자신을 지탱하는 것이야말로 일을 하는 데 있어서 주요한 동기 중 하나입니다. 커리어 성장, 개인적인 성취도 및 기쁨과 더불어 수입 잠재력 향상은 더 발전할 수 있는 원동력이 됩니다. 다른 사람과 공유할 수 있는 성공 스토리를 반드시 준비하세요. 하지만 유념할 것은 하루아침에 완성되는 과정이 아니라는 것입니다. 목표를 염두에 두고 날마다 투입하는 모든 노력, 투자, 헌신, 열정의 집합체가 되어야 합니다.

여러분의 브랜드와 스토리를 확립하기 위해 최대한 모든 기회를 활용함으로써 기회가 왔을 때 스토리가 여러분의 가치를 증명하여 성공을 보장할 수 있어야 합니다.

<https://www.redhat.com/sysadmin/increasing-earning-potential>에서 제공하는 Creative Commons 4.0 BY-SA 라이선스에 따라 공개되고 Enable Sysadmin에 게시된 "Linux 시스템 관리자로서 수입 프로필 및 잠재력을 향상하는 6가지 방법"에서 발췌함.



Joseph Tejal

Joseph은 Red Hat의 기술 계정 관리자입니다. 본질적으로는 Unix 시스템 관리자로서 서비스 개선 이니셔티브에 참여하고 Ansible을 사용해 일상적인 태스크를 자동화합니다. 다수의 고객과 함께 자동화 프로젝트 제공 관련 작업을 수행했고 뉴질랜드에서 Ansible 및 OpenShift 모임을 활발하게 공동 주관해 왔습니다.

자동화 우선 마인드로 전환하기 위한 5가지 방법

DevSecOps를 통해 조직의 경쟁력을 강화할 수 있습니다.
이 다섯 가지 전략을 사용해 시작하세요.

작성자: Allen Eastwood(Red Hat Consulting 선임 아키텍트) 및 Larry Spangler(Red Hat Services Portfolio 관리팀 수석 솔루션 매니저)

자동화 우선 마인드는 어떤 조직에서든 상당한 트랜스포메이션으로 간주될 가능성이 높습니다. 일반적으로 태스크 자동화로 시작해 복잡한 워크플로우 오케스트레이션 단계를 거쳐 결국 지능적인 운영 및 “푸시 버튼” 최종 사용자 서비스를 혁신하는 데까지 나아가기 때문입니다. 이는 흔들림 없이 DevSecOps에 매진함을 뜻하며 이러한 유형의 문화적 변화가 제공할 수 있는 경쟁 우위를 인정하는 것입니다. 하지만 그러한 목표에 도달하고 이를 위해 필요한 지원을 찾아 구축하는 것은 정말 어려운 일입니다. 이는 초기에 개별 부서에서 자동화를 실행하는 데 성공한 사례가 있다 하더라도 마찬가지입니다.

초기에 잦은 빈도로 성공하기

처음부터 도메인 내에 상주하는 가장 중요하고 가장 시간이 많이 소요되는 수동 프로세스를 자동화하고 싶은 유혹이 생길 수 있으나 프로세스 초기에 더 작은 성공을 모색하면 많은 장점을 얻을 수 있습니다. 처음에는 더 광범위한 프로세스의 개별 조각들을 자동화함으로써 나중에 더 복잡한 자동화를 위한 구성 요소를 확보할 수 있고 조기 자동화의 가치를 증명할 수 있습니다.

진행은 반복적이어야 하며, 포함할 각 반복과 그 특징을 고려해야 합니다. 마치 개발자의 일을 하고 있는 듯한 느낌이어야 합니다. 범위와 기능을 반복적으로 늘림으로써 더 큰 목표를 달성할 수 있는 역량을 키우는 동시에, 구축하고 테스트한 자동화의 장점을 누릴 수 있습니다. 이는 곧 확신을 갖고 자동화를 추진할 수 있다는 뜻입니다.

이렇게 적시할 수 있는 작은 성공들은 팀, 상사, 다른 팀에게 자동화 이니셔티브를 설득하는 데 도움이 됩니다.

기술 향상 교육 및 타사 지원 수용

교육 및 자격증은 자동화 우선 마인드로의 전환과 관련된 모든 측면에 매우 중요합니다. 팀이 확신을 갖고 자동화를 배포할 수 있도록 지원하는 데 있어서 핵심이 되는 요소일 뿐 아니라 조직 전반에서 더 폭넓게 자동화를 도입할 수 있는 추진력을 형성하는 데도 도움이 됩니다. 다른 팀들의 역량을 강화하는 동시에 확립된 자동화 표준 및 거버넌스에 대한 컴플라이언스를 보장하는 “강사 교육(train-the-trainer)” 접근 방식의 장점을 누릴 수 있습니다.



많은 조직이 컨설팅 서비스 범위 밖에서 전략적으로 접근하여 **자동화 도입 프로세스를 가속화**합니다. 이는 조직 내 IT 전문가들을 불안하게 만들 수 있습니다. 전문가들은 이것이 영구적인 아웃소싱 또는 실업을 의미할까봐 두려워합니다. 하지만 양질의 컨설팅 서비스는 멘토링 관계를 통해 이러한 팀을 지원하고 팀의 역량을 강화하는 측정 가능하고 유한한 목표를 갖고 있습니다. 이로써 프로덕션 단계의 자동화와 갈수록 복잡해지는 프로젝트 및 활용 사례를 지속적으로 개발할 수 있는 역량을 팀에 부여합니다.

지지자 찾기

대규모 변경은 벅찬 일일 수 있으며, 엔터프라이즈 자동화와 같은 이니셔티브에는 이러한 유형의 트랜스포메이션에 수반되는 자연스러운 저항을 오히려 극복할 수 있는 기회로 즐기는 사람이 요구됩니다. 이러한 사람은 조직에서 비교적 신참에 속하는 경우가 많으며 “미개척지로 나아가” 문제를 해결하라는 지시를 종종 받습니다. 바로 여러분일 수도 있습니다. 어쨌든 이들은 아이디어를 실행하고 이를 개발 및 운영 등에서 긴밀히 협력하기를 꺼릴 수 있는 팀도 포함하여 팀 전체에 확산하는 데 필요한 동력입니다.

자동화 우선 마인드를 확립하려면 협업이 필요합니다.

거버넌스 및 표준 확립

자동화 우선 마인드는 새로운 태스크 및 프로세스를 기꺼이 자동화하고자 하는 마음가짐입니다. 이로써 정보 소스가 확립되어 다른 팀이 여러분의 작업을 통해 안전하게 혜택을 얻을 수 있도록 향후 작업 과정이 문서화됩니다. 여러분은 코드와 변경 사항을 추적(자동화 코드를 정보 및 데이터 모델에서 분리)하여 신뢰할 수 있는 톨로 권한 있는 정보의 보안을 유지하고 소스 코드 분산을 위한 파일 공유를 방지할 수 있는 방법을 원할 것입니다. 코드를 도큐멘테이션으로 사용함으로써 협업을 증진할 뿐 아니라 전사적으로 공유할 수 있는 반복 가능한 자동화 사례를 갖추도록 조직을 혁신할 수 있습니다.

자동화 코드를 중앙에서 관리하는 것 외에도 표준을 정의하고 발전시켜 팀과 프로젝트 전반에서 더 즉각적으로 자동화를 통합하고 오케스트레이션하는 것이 필수적입니다. 광범위한 도입과 재사용이 증가함에 따라 다양한 팀에 해당되는 과제를 해결하기 위한 관리 그룹 설립을 고려해 보세요. 여기에는 일반적으로 표준 톨링, 자동화된 테스트에 대한 요구 사항, 변경 사항 배포 또는 롤백을 위한 모범 사례가 포함됩니다.

커뮤니티를 조성하여 협업

자동화 우선 마인드를 확립하려면 협업이 필요합니다. 이러한 현실로 인해 많은 사람들이 흥미를 잃고 자동화에 회의적일 수 있습니다. 하지만 DevSecOps의 핵심은 재미가 있든 없든 이 팀들을 한자리에 모이게 하는 것입니다. 이 프로세스가 마법처럼 모든 문제를 해결하는 것은 아니지만 통합 표준과 자동화 관리 프로세스를 포함한 복잡한 오케스트레이션이라는 결과를 얻을 수 있습니다. 이러한 자동화는 조직의 실제 고충에 초점을 맞춰야 하며 활동 중인 팀에 실질적이고 측정 가능한 비즈니스 가치를 제공해야 합니다.

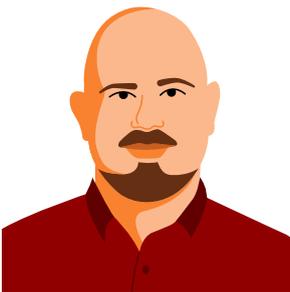


협업이 더 빠르고 효율적으로 진행되게 하는 한 가지 방법은 조직 내에 자동화 커뮤니티를 조성하는 것입니다. 이 커뮤니티는 핵심적인 자동화 팀이 되어야 하며 조직 전체의 인력을 포괄해야 합니다. 이들의 사명은 접근 방식을 공유하고, 과제를 해결하고, 다른 팀들이 자동화를 성공적으로 도입할 수 있게 지원하는 것입니다. 이들은 다른 팀들을 참여시키고, 팀들에게 성공 사례와 가능성을 보여주고, 표준 사례에 관해 안내하며, 다른 사람들이 문제를 해결하고 자동화 사용을 가속화하는 데 도움이 되는 지식(및 플레이북)을 전체적으로 공유함으로써 이러한 사명을 이행합니다.

지향하는 목표는 무엇인가요?

자동화는 비즈니스 영향에 초점을 둘 때 가장 강력합니다. 이러한 영향은 내부 생산성(예: 푸시 버튼 환경) 또는 고객 대상(예: 더 빠른 고객 대응 시간)일 수 있습니다. 조직 내에서 자동화가 더 광범위하게 도입됨에 따라 발전을 계속 추진하고 가치를 증진하기 위해선 워크플로우 및 기본적인 접근 방식을 혁신해야 한다는 점을 알게 될 것입니다. 예를 들어, 프로세스 중간에 승인이 내려지던 프로세스는 완전히 자동화하기 위해 재구성됩니다. 자동화할 태스크를 엄격한 기준에 따라 찾기보다는 수동 태스크를 자동화된 태스크로 교체하는 것을 넘어 워크플로우 및 프로세스를 지속적으로 최적화할 수 있는 플랫폼을 구축하는 것입니다. 이벤트 기반의 자동화된 문제 해결을 수행하고, 다운타임 및 고객 서비스 중단을 최소화하는 패치 적용을 통해 컴플라이언스를 자동화하고, 더 안전한 방식으로 혁신을 진행하여 인간의 과도한 개입을 없애는 프로세스 및 워크플로우를 설계할 수 있는 플랫폼을 구축합니다.

자동화는 근본적으로 간단한 솔루션이지만 개방성이 뛰어나므로 신속하게 변경할 수 있습니다. 자동화는 특성상 비즈니스 요구 사항에 맞게 조정할 수 있는 유연한 솔루션이므로 여러분의 목표는 신속하게 바뀔 것입니다. 최종 결과는 자동화 우선 관점이지만 해결책 자동화, 반복 및 최소 기능 제품(MVP), 작업 최적화는 여러분의 역량과 진화하는 비즈니스 요구 사항에 맞춰 조정되어야 합니다.



Allen Eastwood

Allen은 20년 이상의 경험을 보유한 Red Hat Consulting 선임 아키텍트입니다. 고객의 IT 시스템 자동화를 지원하지 않을 때는 아이스하키를 즐깁니다.



Larry Spangler

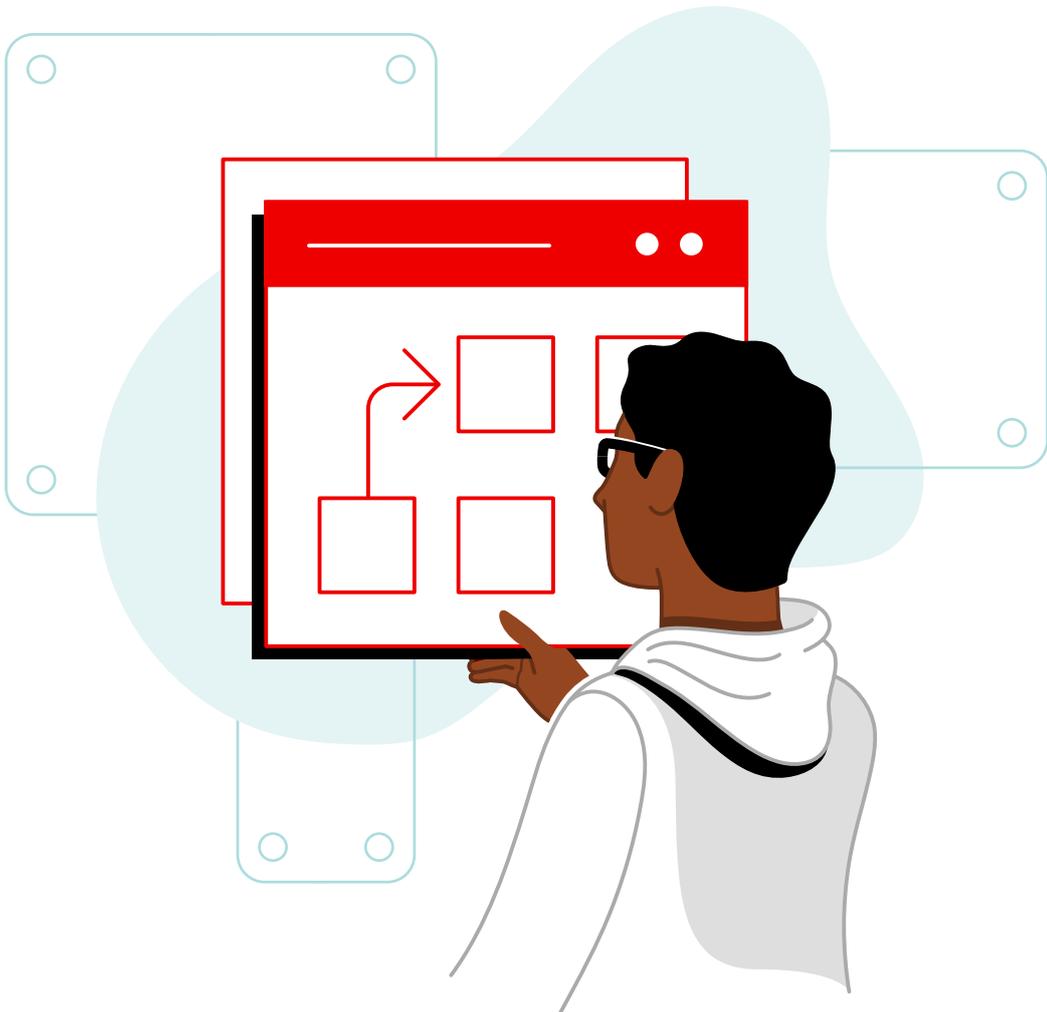
Larry는 Red Hat Services Portfolio 관리팀의 수석 솔루션 매니저로서, Red Hat의 글로벌 서비스 조직을 위한 자동화 서비스 제품 정의, 개발, 관리를 주도합니다. IT 소프트웨어 개발, 시스템 관리, 전문 서비스 분야에서 30년 이상 경험을 쌓은 그는 폭넓은 경험과 인사이트를 사례에 적용합니다.

결론

IT 자동화를 시작하는 방법

45 IT 자동화 수용: 커리어에 도움이 되는 이유와 시작하는 방법

48 시작하기



IT 자동화 수용: 커리어에 도움이 되는 이유와 시작하는 방법

Red Hat은 즉각적인 비즈니스 영향으로 변환되는 통합 교육 옵션으로 자동화를 더 효율적으로 진행할 수 있도록 지원합니다.

작성자: James Mighion(Red Hat Global Learning Services 자동화 사례 부문 매니저) 및 Steven Bonneville(Red Hat 교육 수석 기술 아키텍트)

2019년 Comcast는 거버넌스 및 제어를 통해 프로젝트와 팀 전반에서 자동화 기능을 생성, 공유, 재사용해야 할 필요가 있음을 깨달았습니다. 또한 이러한 노력을 관리할 사내 전문성이 필요했으므로 Comcast는 Red Hat과 협력하여 5일 동안 실시된 온사이트 교육에 참여했습니다. Comcast의 음성 엔지니어들은 전문가가 안내하는 핸즈온 커리큘럼으로 IT 직원들에게 자동화를 구현 및 관리하고 자동화 관련 문제를 해결할 수 있는 기회를 제공하는 Red Hat Ansible Automation Platform 관련 Red Hat 교육 과정을 수강했습니다.

하지만 개인 차원에서도 Red Hat 교육 및 자격증의 장점을 얻을 수 있습니다. 많은 IT 전문가들이 자동화에 관한 교육을 스스로 찾고 있으며 교육이 소속 회사의 가치와 본인의 시장 가치를 높인다고 생각합니다. 2021년 Red Hat Certified Professional of the Year로 선정된 IT 전문가 Christian Sandrini를 예로 들 수 있습니다. 그는 Red Hat Certified Specialist in Ansible Best Practices, Red Hat Certified Specialist in Ansible Automation Platform 등 10여 개의 Red Hat 자격증을 취득했습니다.

Sandrini는 Red Hat 교육 서브스크립션 가입자로서 Red Hat 교육에 적극적으로 참여하고 있습니다. 그는 이렇게 말합니다. “Red Hat 교육 서브스크립션은 제 학습 방식과 맞아서 많은 도움이 되었습니다. 어떤 동영상을 다시 보거나 핸즈온 성격이 강한 랩을 수행할 수 있는 자기 주도식 학습 플랫폼을 유용하게 활용하고 있습니다. 이 플랫폼을 통해 기술을 매우 빨리 익힐 수 있어서 특정 제품에 대해 알지 못한다 하더라도 클래스에 등록하여 해당 제품이 어떤 것인지 신속하게 알아볼 수 있습니다. 또 한 가지 정말 좋은 점은 Red Hat 교육 서브스크립션의 액세스 우선권 섹션입니다. 향후 출시될 기술을 미리 살펴볼 수 있기 때문입니다.”

Sandrini가 자신의 지식을 구현하기 위해 내디딘 첫걸음은 조직에 Red Hat Ansible Automation Platform을 소개하는 것이었습니다. 그는 코드형 인프라(IaC) 접근 방식으로 기능을 자동화하고 팀이 서버를 프로비저닝하고 구성하는 방식을 현대화하는 것을 목표로 했고, 이러한 작업에 대해 팀으로부터 압도적으로 긍정적인 평가를 받은 후 Ansible Automation Platform을 배포했습니다. Sandrini는 이렇게 말합니다. “이때부터 진정한 변화가 일어나기



시작했습니다. 이를 통해 역할 기반 액세스 제어가 가능해지자 다른 팀들이 관심을 갖기 시작했습니다. 따라서 Ansible은 Linux뿐 아니라 Linux 주위의 다른 구성 요소에도 사용할 수 있음을 알 수 있습니다.”

Sandrini는 Red Hat 교육 및 자격증을 통해 습득한 지식을 사용해 IT 자동화를 구현하고 유지 관리하여 프로세스를 간소화하고 비용을 줄임으로써 소속 팀의 소중한 자산이 되었습니다. 그는 기술을 습득하고 향상할 뿐 아니라 팀원 전체에 지식을 나누기 위해 여러 팀에게 교육을 제공해야 한다고 주장합니다.

20년이 넘는 기간 동안 Red Hat은 기술 전문가가 현재 및 새로운 동향과 보조를 맞출 수 있도록 교육 및 자격증 옵션을 제공해 왔습니다.

IT 자동화 기술이 뛰어난 전문가에 대한 수요가 계속 증가함에 따라 새로운 전략과 기술의 동향을 파악하는 것의 중요성 역시 커지고 있습니다. 갈수록 진화하는 IT 분야에서 교육과 전문성 개발은 개인 및 조직 차원에서 경쟁력을 유지하기 위한 핵심 요소입니다. Red Hat 교육 및 자격증은 산업과 보조를 맞추기 위해 지속적으로 교육 과정을 업데이트함으로써 IT 자동화 분야에서 커리어를 향상하고자 하는 교육 받은 공인 전문가들이 성공에 필요한 수단을 갖출 수 있도록 보장합니다.

20년이 넘는 기간 동안 Red Hat은 기술 전문가가 현재 및 새로운 동향과 보조를 맞출 수 있도록 교육 및 자격증 옵션을 제공해 왔습니다. Red Hat 교육 과정에는 전통적인 강의실 기반 교육에서 온사이트 교육, 다양한 가상 학습 옵션에 이르기까지 다양한 요구 사항 및 학습 스타일에 적합한 여러 옵션이 마련되어 있습니다. 평균적으로 20년의 IT 분야 경력과 11년의 Red Hat 제품 사용 경력을 보유한 Red Hat 공인 강사진은 전문성을 통해 비즈니스 목표에 즉각적인 영향을 미치기 위해 노력합니다.

Red Hat 교육 서브스크립션을 통해 학습자는 자신의 기기에서 실시간 가상 교육을 이수하고 자기 주도식 과정, 동영상, 랩의 전체 카탈로그에 즉시 액세스할 수 있습니다. 이 모든 콘텐츠는 산업과 보조를 맞추기 위해 지속적으로 업데이트됩니다. 실제로 Red Hat 교육 서브스크립션 카탈로그에서 제공하는 과정의 41%는 지난해에 추가되거나 업데이트되었습니다. 일부 Red Hat 교육 서브스크립션 티어에는 Red Hat 자격증 시험 응시 비용도 포함되므로 지식을 습득하고 지식 수준을 증명하기까지의 여정이 간소화됩니다.

Red Hat 교육은 수강생이 자동화 및 디지털 트랜스포메이션 분야의 최신 동향에 적응하도록 지원하고, Red Hat 자격증은 IT 전문가가 진화하는 산업이 제시하는 과제에 직면하여 가장 야심 찬 프로젝트를 수행할 기술과 준비를 갖추고 있는지 검증합니다. Red Hat은 현재 IT 자동화에 중점을 두고 있는 두 가지 자격증을 제공하고 있습니다. Red Hat Enterprise Linux Automation with Ansible(RH294)과 Red Hat Certified Engineer(RHCE®) 시험은 더 효율적인 개발을 위한 워크플로우 자동화, DevOps 사례 활용, Ansible Automation Platform 사용의 기초를 쌓는 과정입니다. 심화 과정인 Advanced Automation: Red Hat Ansible Best Practices(DO447)와 Red Hat Certified Specialist in Advanced Automation: Ansible



Best Practices 시험은 대규모 엔터프라이즈 환경의 여러 사업부 전반에서 기존 Ansible 인프라를 사용하고 확장하는 데 필요한 기술을 발전시키고 입증합니다. Red Hat Certified Professional은 자동화 분야 전문성을 통해 조직이 최적의 효율성과 비용 절감을 실현하도록 지원합니다. 또한 조직은 온보딩 시간 단축 및 비용 절감 확대에 의해 공인 전문가를 채용하고 유지하고자 합니다.

Steven Bonneville



Steven Bonneville은 Red Hat의 수석 기술 아키텍트로서 Red Hat 교육 부서에 몸담고 있습니다. 20년 넘게 Red Hat에서 근무한 Steven은 Red Hat 기술에 대한 깊은 전문성을 통해 Red Hat의 교육 커리큘럼을 위한 과정을 설계하고, 커리큘럼 개발자의 멘토가 되어 이들을 지도하고, 향후 교육 개발의 방향을 제안합니다. 수년 동안 그는 RHCE 교육 과정과 시스템 관리, 가상화, 스토리지에 관한 고급 과정을 포함한 Red Hat Enterprise Linux 시스템 관리 커리큘럼을 담당해 왔으며 이러한 과정 중 다수의 최초 버전을 작성했습니다. 현재는 Red Hat Ansible Automation Platform을 사용해 플랫폼 간 자동화 교육을 개발하는 데 집중하고 있습니다.

James Mighion



James Mighion은 Red Hat의 Global Learning Services 조직에서 자동화 사례 매니저로 근무하고 있습니다. 2011년 이후 Red Hat과 함께하면서 다양한 역할을 맡아 온 James는 자동화에 열정을 갖고 있으며 오픈소스 프로젝트에 기여하고 있습니다.

시작하기



Red Hat Ansible Automation Platform 체험판 시작하기

Red Hat Enterprise Linux용 Red Hat Ansible Automation Platform 단일 셀프 지원 60일 서브스크립션을 사용해 보세요.

→ [60일 동안 Ansible Automation Platform 체험하기](#)



Red Hat Ansible Automation Platform의 기본 사항 학습하기

온디맨드 온라인 동영상 시리즈를 시청하세요. Ansible Essentials: Simplicity in Automation Technical Overview(DO007)는 클라우드, 가상 및 물리 환경 전반의 컴퓨팅 인프라 구성 관리, 프로비저닝, 배포, 관리를 포함해 Ansible Automation Platform에 대해 소개합니다.

→ [무료 교육을 통해 Red Hat Ansible Automation Platform에 대해 알아보기](#)



Enable Sysadmin - 시스템 관리자를 위한 커뮤니티

Enable Sysadmin은 가이드, 방법, 튜토리얼, 설명, 팁과 요령 등을 널리 알리는 커뮤니티 출판물입니다. 이 출판물에서는 시스템 관리자가 자주 사용하는 기존 및 이머징 기술 개념, 소프트웨어 및 애플리케이션, 스크립팅 및 프로그래밍 언어와 일반적인 과제 및 솔루션에 대해 다룹니다.

→ [뉴스레터 구독하기 | 커뮤니티에 가입하기](#)



자동화로 혁신 선도, 자동화를 통한 소통

디지털 리더들은 기존 워크플로우를 자동화하는 데 그치지 않고, 자동화를 이용해 공유 가치를 창출하는 방법을 배우고 있습니다. 여러분의 조직에서도 이미 IT의 일부분을 자동화하고 있을 것입니다. 하지만 여러분은 고급 단계에 있으신가요, 아니면 시작 단계인가요? 온라인 평가를 통해 현재의 자동화 성숙도에 대해 알아보고, 다음 단계를 파악하고, 진행 과정에 도움이 되는 리소스도 얻으세요.

→ [평가하기](#)