Zero dynamics를 이용하여 제어시스템을 몰래 공격하는 방법

심형보 (서울대학교 전기정보공학부)

제어이론을 잘 알면 제어시스템의 보안을 뚫고 시스템을 파괴하는 일도 할 수 있게 된다. 이번 호에서는 제어시스템 공격 방법에 대해 알아보고자 한다. 특별히 zero dynamics가 무엇인지 알아보고 zero dynamics attack이라고 알려진 무서운 공격 방법에 대해 알아볼 것이다. 물론 이 기사의 목적은 해커를 양성하고자 함이 아니라 이러한 공격 방법을 이해하고 이를 방어하기 위한 노력이 필요함을 역설하고자 함이다.

1. 제어시스템의 보안 문제

제어시스템 보안이란 무엇을 말하는 것일까? 널리 알려진 컴퓨터 보안, 혹은 사이버 보안이나 정보 보안이라고 일컬어지는 것의 의미를 찾아보면, '하드웨어, 소프트웨어 또는 데이터의 도난이나 손상, 컴퓨터가 제공하는 서비스의 중단 또는 오용으로부터 컴퓨터 시스템을 보호하는 것'이라고 되어 있다. 이 경우 해커[1]의 목적은 데이터의 도난이나 손상, 그리고 컴퓨터 시스템의 오작동인 것이다. 반면, 제어시스템 보안, 혹은 사이버–물리–시스템(Cyber-Physical System) 보안이라고 하는 것은, 해커로부터 제어의 대상인 플랜트(plant; 물리시스템)의 동작을 보호하는 것이라고 할 수 있다. 이 경우 해커의 목표는 플랜트의 오작동 혹은 손상이라 하겠다.

해커는 어떻게 플랜트의 통제권을 얻게 될까? 가장 손쉬운 방법은 제어기 컴퓨터에 침투하여 해당 컴퓨터의 통제권을 얻는 것이다. 다만, 컴퓨터 보안이 잘 되어 있어 (컴퓨터 공학자가 일을 열심히 했다고 하자), 이와 같은 방법이 여의치 않을 때 해커는 통신 네트워크에 침투하는 대안을 생각할 것이다. 이는 그림 1에서와 같이 플랜트의 센서단에서 측정된 플랜트의 출력 정보가 통신 네트워크를 통해 제어기에 전달되고, 연산을 통해 결정된 제어 입력이 다시 통신 네트워크를 통해 구동기단에 전달되는 경우에 해당한다. 만일 해커가 신호 전달 경로에 개입하게 되면, 실제 측정된 신호 y 대신 자신이 만든 신호를 제어기에 전달할 수 있고, 또한 실제 제어기가 생성한 제어 입력 u 대신 자신의 신호를 구동기에 전달할 수 있게 된다. 이 경우, 해커는 센서 신호와 제어 신호를 교란할 수 있도록 통신 네트워크의 통제권을 확보한 상태에서 공격 개시 시점을 기다릴 것이다. 이러한 상황을 용이하게 묘사하기 위해 그림 1에 보는 것처럼 해커는 자신의 신호 a_a와 a_s를 각각 u와 y에 마음대로 더할 수 있는 것으로 기술하였다. 즉, 평시에는 $a_a(t) = a_s(t) = 0$을 유지하다가, 원하는 공격 시점에 이들 신호를 주입할 수 있다는 뜻이다.

한편, 해커는 자신이 공격을 개시한 시점으로부터 실제 플랜트가 손상되기까지 약간의 시간이 필요함을 알고 있으며, 이 시간동안 자신의 공격이 제어기로부터 발각되지 않기를 원할 것이다. 왜냐하면 원자력 발전소나 철도 등 중요한 제어시스템의 경우 시스템의 이상 발생시 즉각적으로 기존의 제어를 멈추고 플랜트를 보호하는 부가 시스템이 작동하게 되므로, 공격이 발각되어 시스템의 이상 발생으로 판단되는 즉시 공격은 실패한 것으로 간주해야 하기 때문이다. 따라서 해커는 많은 경우 센서 신호 y와 제어 신호 u를 완전히 무시하고 자신의 신호를 주입하기 보다는, 이들 신호가 제어기와 구동기에 각각 전달되어 제어기의 관점에서 시스템이 어느 정도 정상적으로 동작하는 것처럼 보이게 하는 동시에, 자신의 신호 a_a와 a_s를 영리하게 설계하여 공격이 발각되는 시점을 늦추면서 플랜트를 손상시키기를 희망할 것이다.

이번 호에서 제어이론을 공부한 해커가 과연 얼마나 영리하게 이러한 공격 신호를 설계할 수 있는 지 알아보자.

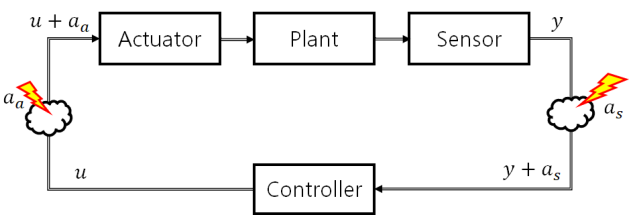

Fig. 1. 그림에서 a_a와 a_s는 각각 해커가 통신 네트워크에 주입하는 actuator attack signal과 sensor attack signal을 의미한다.

[1]사실 hacker는 종종 컴퓨터 전문가를 뜻하기 때문에, cracker, attacker, 혹은 adversary라고 표기해야 하나, 편의상 해커라 하겠다.

2. 맛보기 : REPLAY ATTACK

제어시스템의 보안 문제에 대한 관심이 증대함에 따라 IEEE Control Systems Magazine에서 특집호가 발간되기도 하였는데 관심있는 독자는 [1]을 참고해 보면 좋을 것이다. 사실 제어의 대상이 국가기반시설이 될 경우, 해킹의 결과는 개인정보 탈취나 경제적인 손실의 수준을 넘어 사람의 생명이 손상될 정도로 심대하다 보니 최근들어 그 중요성이 날로 증가하고 있다.

본 편의 본격적인 논의에 앞서 [1]에 소개된 간단한 공격 방법을 하나 소개하고자 한다. 이는 replay attack이라 불리는 것으로, 정상적으로 동작하는 제어시스템의 센서 신호 $y(t)$를 일정 시간 T동안 $y_{\text{rec}}(t)$으로 기록한 뒤, 공격이 t_0 시간에 시작되면 $a_s(t) = y_{\text{rec}}(t-t_0) - y(t)$를 인가하고, 매 T 시간마다 이 방식을 반복하는 것이다. 이를 통해 제어기는 실제 플랜트에서 무슨 일이 벌어지고 있는 지 알지 못하게 되고, 이 시간동안 $a_a(t)$를 이용해 플랜트가 손상될 수 있는 제어 입력을 인가하면 된다. 이러한 공격은 마치 은행털이범이 CCTV에 정상적으로 녹화된 장면을 송출하는 동안 범죄를 저지르는 것과 유사해 replay attack이란 이름을 갖게 되었다. 다만 이 공격은 쉽게 회피가 가능한데, 예를 들면 제어입력에 약간의 watermarking 신호를 포함한 뒤 제어기에 되돌아 온 센서 신호를 분석해 replay attack 유무를 판별할 수 있다 [1].

3. 제어이론 공부

3.1. Normal Form

본격적으로 zero dynamics attack을 소개하기 위해, 기본적인 선형시스템 공부를 잠시 해 보자. 선형시스템 과목에서는 보통 controllable canonical form, observable canonical form, Jordan form 등 상태변수 표현법의 몇가지 기본형을 소개한다. 본 절에서 소개할 normal form도 이러한 기본형의 일종이라 할 수 있는데, 어떤 단일입력-단일출력 시스템이 아래의 형태로 주어질 때 '이 시스템은 normal form[2]으로 표현되었다'고 한다:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
&\vdots \\
\dot{x}_{\nu-1} &= x_\nu \\
\dot{x}_\nu &= \phi x + \psi z + gu \\
\dot{z} &= Sz + Gx \\
y &= x_1.
\end{aligned} \tag{1}$$

여기서, ν는 이 시스템의 상대차수(relative degree)라고 하는데, 이 선형시스템을 전달함수로 표현했을 때 분모 다항식의 차수에서 분자다항식의 차수를 뺀 것으로 정의된다. 이는 입력 u가 출력 y에 영향을 미치기까지 적분기를 몇 개 통과해야 하는가를 나타내는 값이기도 하다. 변수 x_i는 모두 scalar이고 $x = [x_1, x_2, \cdots, x_\nu]^T$이며, 변수 z는 $n-\nu$개의 원소를 갖는 열벡터이다. 따라서 이 시스템의 상태변수를 $\xi \in \mathbb{R}^n$라 하면, $\xi = [x^T, z^T]^T$가 되는 셈이다. 또한, $\phi \in \mathbb{R}^{1\times\nu}$, $\psi \in \mathbb{R}^{1\times(n-\nu)}$, $g \in \mathbb{R}^{1\times 1}$, $S \in \mathbb{R}^{(n-\nu)\times(n-\nu)}$, 그리고 $G \in \mathbb{R}^{(n-\nu)\times\nu}$는 모두 적절한 크기를 갖는 행렬이다.

한가지 주목할 점은, 상대차수가 1 이상인 어떤 단일입력-단일출력 선형 시스템

$$\begin{aligned}
\dot{\chi} &= A\chi + bu, \\
y &= c\chi
\end{aligned} \tag{2}$$

이 normal form으로 표현되지 않았을 때, 우리는 언제나 적절한 좌표변환 $\xi = T\chi$를 통해 변환된 시스템

$$\begin{aligned}
\dot{\xi} &= TAT^{-1}\xi + Tbu, \\
y &= cT^{-1}\xi
\end{aligned} \tag{3}$$

이 normal form (1)이 되도록 할 수 있다는 것이다. 이를 보이기 위해 우선 $cA^{\nu-1}b \neq 0$이고, $i = 0, \cdots, \nu-2$에 대해 $cA^ib = 0$임에 주목하자. 이는 시스템의 상대차수가 ν이기 때문인데, 출력 y는 그것의 $\nu-1$번째 미분까지 입력 u가 명시적으로 보이면 안되기 때문에, $\dot{y} = cAx + cbu$로부터 $\nu > 1$이라면 $cb = 0$임을 알 수 있고, 이에 따라 $\ddot{y} = cA^2x + cAbu$라는 식에서 $\nu > 2$라면 $cAb = 0$일 수 밖에 없다. 이와 같은 논리로, $i = 0, \cdots, \nu-2$에 대해 $y^{(i)}$의 전개에서 u가 보이지 않기 때문에 $cA^ib = 0$임을 알 수 있고, $y^{(\nu)} = cA^\nu x + cA^{\nu-1}bu$에서는 u가 등장해야만 하기 때문에 $cA^{\nu-1}b \neq 0$임을 알게 된다.

이로부터

$$\begin{bmatrix} c \\ cA \\ \vdots \\ cA^{\nu-1} \end{bmatrix} \begin{bmatrix} b & Ab & \cdots & A^{\nu-1}b \end{bmatrix}$$

$$= \begin{bmatrix} 0 & \cdots & 0 & cA^{\nu-1}b \\ 0 & \cdots & cA^{\nu-1}b & * \\ \vdots & \vdots & \vdots & \vdots \\ cA^{\nu-1}b & * & * & * \end{bmatrix}$$

[2]Normal form은 비선형제어이론에서 주로 사용되었고 처음 사용한 사람들의 이름을 붙여 Byrnes-Isidori normal form이라고도 한다. Normal form이란 용어가 여러 분야에서 서로 다른 의미로 사용되기 때문에 혼란을 방지하려는 목적도 있다.

라는 식이 성립함을 알 수 있는데, 이 식의 우변 행렬이 nonsingular라는 사실은 $cA^{\nu-1}b \neq 0$으로부터 자명하고, 따라서 좌변의 두 정방행렬도 모두 nonsingular라는 것을 알게 되며, 이로부터 행벡터인 cA^i는 $i = 0, \cdots, \nu-1$에 대해 모두 선형독립임을 알 수 있다. 그렇기 때문에, $\Phi b = 0$을 만족하는 동시에 행렬

$$T = \begin{bmatrix} c \\ cA \\ \vdots \\ cA^{\nu-1} \\ \Phi \end{bmatrix} \in \mathbb{R}^{n\times n}$$

이 nonsingular하도록 만드는 행렬 $\Phi \in \mathbb{R}^{(n-\nu)\times n}$를 항상 찾을 수 있다. 왜냐하면, 행렬 $b \in \mathbb{R}^{n\times 1}$의 left-null space의 차수는 $(n-1)$이므로 $n-1$개의 선형독립인 행벡터가 존재해야만 하는데, 이 중 $\nu-1$개의 선형독립인 행벡터 $cA^i (i = 0, \cdots, \nu-2)$는 이미 찾았고, $cA^{\nu-1}$은 $cA^{\nu-1}b \neq 0$이기 때문에 b의 left-null space에 속하지 않으므로, 우리는 여전히 $cA^i (i = 0, \cdots, \nu-1)$과 선형독립이면서 b의 left-null space에 속하는 $n-\nu$개의 행벡터를 찾을 수 있다. 이들 행벡터를 찾아 Φ 행렬의 개별 행으로 두게되면 원하는 행렬 $\Phi \in \mathbb{R}^{(n-\nu)\times n}$이 얻어진다.

이렇게 얻어진 행렬 T를 통해 주어진 시스템 (2)를 좌표변환하면 정말 (1)이 되는지 확인해 보자. 우선, $\xi = [x^T, z^T]^T = T\chi$로부터, $x_1 = c\chi$이고 $x_2 = cA\chi$이므로, $\dot{x}_1 = cA\chi + cbu = x_2$가 되고, 이런 방식으로

$$\dot{x}_i = x_{i+1}, \qquad i = 1, \cdots, \nu-1$$

임을 알 수 있다. 또한, $[\phi, \psi] := cA^{\nu}T^{-1}$라는 식을 이용해 두 행벡터 $\phi \in \mathbb{R}^{1\times\nu}$와 $\psi \in \mathbb{R}^{1\times(n-\nu)}$를 정의하고 $g := cA^{\nu-1}b$라 한다면,

$$\begin{aligned} \dot{x}_\nu &= cA^{\nu-1}(A\chi + bu) = cA^{\nu}T^{-1}\begin{bmatrix} x \\ z \end{bmatrix} + cA^{\nu-1}bu \\ &= \phi x + \psi z + gu \end{aligned}$$

와 같이 된다. 또한, $[S, G] := \Phi AT^{-1}$를 통해 두 행렬 $S \in \mathbb{R}^{(n-\nu)\times(n-\nu)}$와 $G \in \mathbb{R}^{(n-\nu)\times\nu}$을 정의한다면, $z = \Phi\chi$로부터

$$\begin{aligned} \dot{z} &= \Phi A\chi + \Phi bu = \Phi AT^{-1}\begin{bmatrix} x \\ z \end{bmatrix} \\ &= Sz + Gx \end{aligned}$$

가 됨을 확인해 볼 수 있다.

참고 : 시스템 (2)를 (1)로 변환하는 또 다른 방법은 (2)를 전달함수 $c(sI-A)^{-1}b$로 변환하고, 이 전달함수를 (1)의 형태로 realization하는 것이다. 이 과정에 관한 자세한 설명은 [2]의 Example 13.2와 따라오는 본문 설명을 참조하면 된다. (이에 따르면, 식 (1)의 Gx 항은 사실 $G'x_1$과 같이 x_1에만 의존하도록 Φ를 잘 잡을 수 있음이 알려져 있다.) 본 편에 소개된 내용은 좌표변환 행렬 T를 직접적으로 구하는 방법인데, 이 방법은 사실 비선형시스템에 관해 알려진 내용을 가져온 것이다. 단일입력-단일출력 비선형 시스템도 상대차수가 잘 정의될 경우 본 편에 소개된 내용과 같은 방식으로 적절한 비선형 좌표변환을 구해 비선형 normal form의 형태로 변환할 수 있다.

3.2. Normal Form과 Zero Dynamics

그럼 도대체 왜 사람들이 normal form을 중요하다고 하는 것일까? 가장 유용한 점은 시스템의 zero dynamics를 아주 쉽게 볼 수 있다는 점이다.

Zero dynamics란 무엇일까? 이것을 이해하기 위해 먼저 어떤 시스템의 출력 $y(t)$가 모든 시간 $t \geq 0$에서 0이 되기 위해서는 이 시스템의 초기값이 적절히 설정되어야 하는 동시에 입력 $u(t)$가 출력 $y(t)$를 계속해서 0으로 유지하는 신호여야 함을 주목하자. 이러한 입력신호 $u(t)$는 초기값에 따라 달라질 수 있으며, 이러한 초기값과 입력신호의 짝(pair)은 시스템의 성질에 따라 매우 많을 수도 있고 단 한개만 있을 수도 있다. 이 때 이러한 초기값의 집합은 $\mathbb{R}^n$ 공간에서 부분공간을 이루게 되는데, 이 부분공간 상에서 시스템의 출력을 0으로 유지하면서 움직이는 상태변수를 지배하는 미분방정식을 'zero dynamics'라고 한다.

말이 어려웠다면, normal form으로 주어진 (1)에 대해 zero dynamics를 직접 구해보자. 우선, $y(t) = x_1(t)$가 모든 시간동안 0이 되려면, (1)의 구조로부터 모든 시간동안 $x_i(t) = 0\ (i = 1, \cdots, \nu)$여야만 하며, 이를 위해서는 입력신호가

$$u(t) = \frac{-\psi z(t)}{g}$$

여야함을 알 수 있다. 정리하면, 임의의 $z(0) \in \mathbb{R}^{n-\nu}$와 $x(0) = 0 \in \mathbb{R}^{\nu}$으로 만들어진 초기값 $\xi(0) = [x(0)^T, z(0)^T]^T$와 위 입력신호 $u(t)$를 사용하면, 시스템 (1)의 출력 $y(t)$가 0을 유지한다. 이 경우 초기값의 자유도는 $n-\nu$개의 원소를 가진 벡터 $z(0)$이며, 초기값 $\xi(0)$에 입력 $u(t)$를 적용할 경우 시스템 (1)의 $\dot{x}_i(t)$는 모든 시간동안 0으로 고정되고 $\dot{z}(t)$는 이 경우

$$\dot{z} = Sz \tag{4}$$

와 같이 간단하게 되며 바로 이 subsystem (4)가 (1)의

zero dynamics가 되는 것이다.

Zero dynamics란 이름은 출력을 0으로 만드는 것에서 유래된 것이지만, 또한 zero dynamics (4)의 시스템 행렬인 S의 고유치(eigenvalue)가 원래 시스템 (1)의 영점(zero)란 점에 주목해야 한다. 이를 확인하기 위해, (1)의 영점을 구해보자. 주어진 단일입력-단일출력 선형시스템 (2)의 영점을 구하기 위해서는 이 시스템의 Rosenbrock 행렬

$$\begin{bmatrix} \lambda I - A & -b \\ c & 0 \end{bmatrix} \tag{5}$$

이 singular하게 되는 복소수 λ를 구하면 된다. 이를 시스템 (1)에 적용해 보기 위해, S의 고유치와 고유벡터를 각각 λ와 z라 하자. 그러면,

$$\begin{bmatrix} \lambda & -1 & 0 & \cdots & 0_{1\times(n-\nu)} & 0 \\ 0 & \lambda & -1 & \cdots & 0_{1\times(n-\nu)} & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ & & -\phi & & -\psi & -g \\ & & -G & & \lambda I - S & 0 \\ 1 & 0 & 0 & \cdots & 0_{1\times(n-\nu)} & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \bar{z} \\ -\frac{\psi\bar{z}}{g} \end{bmatrix} = 0$$

이 성립하는데, 이 식의 의미는 좌변의 정방행렬이 singular하다는 뜻이고, 이 행렬은 (1)의 Rosenbrock 행렬이므로 λ는 (1)의 영점이 된다.

그러므로, $\dot{z} = Sz$는 시스템 (1)의 zero dynamics이며 S의 고유치는 해당 시스템의 영점이라 보면 된다. 이때 모든 영점의 실수부가 음수이면 이 시스템을 최소위상(minimum phase)이라고 부르고, 그렇지 않으면 비최소위상(non-minimum phase)이라고 부른다.

3.3. Inverse Dynamics

기왕 normal form을 공부했으니 잠시 짬을 내어 normal form의 유용성에 대해 살펴보고 가자. 그 한 예가 시스템의 역동역학(inverse system)을 쉽게 구할 수 있다는 것이다. 예를 들어, 선형 시스템

$$G(s) = \frac{s+1}{s^2+s+1}$$

의 역동역학은

$$\frac{1}{G(s)} = \frac{s^2+s+1}{s+1} = s + \frac{1}{s+1}$$

이므로, 윗 식의 우변을 realization하여

$$\begin{aligned} \dot{x} &= -x + u \\ y &= x + \dot{u} \end{aligned}$$

을 얻으면 이것이 바로 역동역학이 된다. 상대차수가 양수인 전달함수의 분자와 분모를 바꾸면 상대차수가 음수가 되기 때문에, 위 상태변수 표현법에서 출력신호 y가 입력신호의 미분을 포함하는 것은 자연스러운 것이다.

이제 normal form으로 주어진 (1)의 역동역학을 구해보자. 이를 위해, 입력 u와 출력 y를 바꾸어 쓰면, $x_1 = u$가 된다. 또한, 식 (1)로부터 $x_2 = \dot{u}, x_3 = \ddot{u}, \cdots, x_\nu = u^{(\nu-1)}$, 그리고 $y = (1/g)(-\phi x - \psi z + u^{(\nu)})$를 얻는다. 이를 정리하면, $U := [u, \dot{u}, \ddot{u}, \cdots, u^{(\nu-1)}]^T$라 할때,

$$\begin{aligned} \dot{z} &= Sz + GU \\ y &= \frac{-\psi z - \phi U + u^{(\nu)}}{g} \end{aligned}$$

인 $(n-\nu)$차 동역학이 되며, 이것이 바로 (1)의 역동역학이다.

3.4. Tracking Control

기왕 역동역학을 구했으니 한가지 유용한 응용을 살펴보고 가자. 주어진 선형 시스템 (2)의 출력 $y(t)$가 어떤 reference 궤적 $r(t)$를 추종(tracking)하는 궤환제어기(feedback controller)를 구하는 문제를 생각해 보자. 만일 $r(t)$가 또다른 어떤 선형 시스템으로부터 생성되고 이 선형 시스템의 모델을 아는 경우에는 internal model principle을 사용한 output regulator라는 것을 설계하여 적절한 추종제어기를 설계할 수 있다. 하지만, $r(t)$가 선형 시스템으로부터 생성되는 것이 아니라, 임의의 신호인 경우에는 어떻게 추종제어기를 설계할 수 있을까? 이 문제는 $e(t) := y(t) - r(t)$라고 정의하고, $\lim_{t\to\infty} e(t) = 0$을 만족하게끔 궤환제어식을 찾을 수 있으면 해결된다. 이때 $r(t)$가 여러번 미분 가능하다고 하면 normal form은 우리에게 손쉬운 해법을 제시해 준다. 즉, $e = x_1 - r$이고 $\dot{e} = x_2 - \dot{r}, \ddot{e} = x_3 - \ddot{r}$과 같이 되기 때문에,

$$e^{(\nu)} = \phi x + \psi z + gu - r^{(\nu)}$$

임을 알 수 있고, 따라서 어떤 궤환 이득 행렬 $K = [k_1, k_2, \cdots, k_\nu] \in \mathbb{R}^{1\times\nu}$과 $E := [e, \dot{e}, \cdots, e^{(\nu-1)}]^T$에 대해

$$u = \frac{-\phi x - \psi z + r^{(\nu)} - KE}{g} \tag{6}$$

라는 상태변수 궤환제어기를 도입한다면, $R := [r, \dot{r}, \cdots, r^{(\nu-1)}]^T$의 기호를 이용하여 폐루프 시스템을

$$\begin{aligned} e^{(\nu)} &= -KE = -k_1 e - k_2 \dot{e} - \cdots - k_\nu e^{(\nu-1)} \\ \dot{z} &= Sz + GE + GR \end{aligned} \tag{7}$$

라고 쓸 수 있다. 그러므로,

$$s^\nu + k_\nu s^{\nu-1} + k_{\nu-1}s^{\nu-2} + \cdots + k_1$$

이 Hurwitz 다항식이 되도록 K를 설계하면 식 (7)로부터 $\lim_{t\to\infty} e(t) = 0$이 된다. 한편, $r(t)$와 그 미분이 모두 유계(bounded)인 경우에 $E(t)$와 $R(t)$는 모두 유계인 신호가 되고, 행렬 S가 안정(Hurwitz)한 행렬이라면, $z(t)$도 유계인 신호가 되어 제어입력 $u(t)$가 발산하지 않고 내부 상태변수도 모두 유계인 신호가 된다. 따라서, 식 (6)이 구하고자 하는 추종제어기임을 알 수 있다.

4. ZERO DYNAMICS ATTACK

오랜 기다림 끝에 드디어 zero dynamics attack을 살펴볼 수 있게 되었다. 이를 위해 set-point regulation의 목적을 수행하는 출력궤환 제어기가 설계되어 그림 1과 같은 구성을 갖고, 공격 신호 a_a와 a_s가 없을 때 페루프 시스템이 안정한 경우를 생각해 보자. 이 때 플랜트가 (2)와 같이 주어지고 상태변수가 적절히 선택되어 정상상태에 이르러 $\chi(t)$는 0이라 하자. 이 플랜트를 normal form (1)로 표기해도 달라지지 않고 해당 제어기의 동작에 의해 상태변수 $x(t)$와 $z(t)$는 0으로 수렴하게 된다.

그런데, 해커가 구동기단에 공격신호 a_a를 주입할 수 있게 되어 플랜트에 대한 입력신호가 $u(t)$가 아닌 $u(t)+a_a(t)$가 되었다고 할 때, 과연 해커는 어떤 신호 a_a를 주입해야 들키지 않고 플랜트를 공격할 수 있을까? 일단 아래와 같이 생성된 공격신호 a_a를 생각해 보자:

$$\begin{aligned} \dot\eta &= S\eta, \qquad \eta(0) \neq 0, \\ a_a &= -\frac{1}{g}\psi\eta. \end{aligned} \tag{8}$$

여기서 공격시작 시간을 0이라 가정하였다. 이 신호가 주입될 경우, 페루프 시스템에 대한 영향을 살펴보기 위해 z 대신 $e := z-\eta$라는 상태변수를 사용하여, 해커가 만든 동역학 (8)을 포함한 전체 시스템을 써 보면

$$\begin{aligned} \dot x_1 &= x_2 \\ &\vdots \\ \dot x_\nu &= \phi x + \psi z + g(u+a_a) = \phi x + \psi e + gu \\ \dot e &= Se + Gx \\ y &= x_1 \\ \dot\eta &= S\eta, \qquad \eta(0)\neq 0 \end{aligned} \tag{9}$$

과 같이 된다. 이 식을 잘 보면 입력 u와 출력 y간의 관계는 원래 시스템 (1)과 완전히 동일하다는 것을 알 수 있다. 즉, 해커의 공격을 당한 시스템이 원래 시스템과 완전히 동일한 입출력 관계식을 가진다는 것이다! (Zero dynamics의 상태변수 z가 새로운 상태변수 e로 바뀐 형태일 뿐이다.) 출력궤환 제어기는 이 플랜트의 내부 상태변수를 알지 못하고 입력과 출력의 관계만으로 제어를 하기 때문에, 출력궤환 제어기의 동작으로 인해 윗 식의 모든 상태변수 $x(t)$와 $e(t)$는 0으로 수렴하게 된다. 그런데 $e(t) = z(t) - \eta(t)$가 0으로 수렴한다는 것은 결국 $z(t)$가 $\eta(t)$로 수렴한다는 말이고, 만약 S가 불안정한 행렬이고 초기값 $\eta(0)$가 S의 불안정한 모드를 야기하게 설정[3]되었다면 $\eta(t)$는 무한대로 발산할 것이다. 따라서 플랜트의 상태변수의 일부인 $z(t)$ 역시 무한대로 발산하면서 플랜트를 손상시키게 된다. 이것이 zero dynamics attack의 전모이다. 이 과정에서 플랜트의 입출력은 정상과 동일하기 때문에, 이 공격을 검출하는 것은 원리적으로 불가능하다.

하지만 몇 가지 주목할 점이 있다. 우선 행렬 S가 안정한 경우, 즉 최소위상 시스템인 경우에는 이러한 공격이 검출되지는 않겠지만 큰 효과가 없다. 왜냐하면 $\eta(t)$가 0으로 수렴하기 때문이다. 따라서 zero dynamics attack은 비최소위상 시스템에만 효과적인 공격방법이다. 두번째 주목할 점은, 정상상태에 놓여 있는 제어시스템에 공격을 시작할 때 해커 동역학의 초기값 벡터 $\eta(0)$는 절대값이 작은 원소들로 이루어져 있어야 한다. 이유를 알기 위해 공격 전과 후 모두 식 (9)로 플랜트를 표기해 보자. 공격 전에는 $e = z$이기 때문에 $t < 0$에서 정상상태에 이르러 $e(t) = z(t) \approx 0$라고 볼 수 있는데, $t = 0$에서 공격이 시작되면서 갑자기 e의 값이 $e(0) = z(0) - \eta(0)$로 변하게 되고, 이 변화가 크다면 잠시 후 출력 $y(t)$에서 이 변화를 검출하게 될 것이다. 물론, 이론적으로는 시스템이 observable하므로 큰 변화뿐 아니라 아주 작은 변화라도 $y(t)$에 검출되겠지만, 실제 상황에서는 측정센서의 잡음 등으로 인해 상태변수의 작은 변화를 출력에서 검출하기는 쉽지 않고, 해커는 이 점에 착안해서 $\eta(0)$의 원소의 절대값을 매우 작게 설정할 것이다. 이렇다 하더라도 S의 불안정한 모드를 야기하는 초기값 $\eta(0)$에 대해 $\eta(t)$는 결국 무한대로 발산할 것이라 공격의 효과는 동일하다.

한편, 지금까지 소개한 zero dynamics attack을 실제 시스템에 시뮬레이션한 결과를 보고 싶은 독자는 [3]를 참조해 볼 수 있다.

[3] 행렬 S의 고유치 중 실수부가 0보다 같거나 작은 것에 해당되는 고유벡터들로 span된 부분공간을 제외한 곳에 초기값이 위치하면 된다. $\mathbb{R}^n$ 공간에서 차수가 n보다 작은 부분공간의 부피(measure)는 0이기 때문에 무작위로 선정된 초기값이 그 부분공간에 놓일 확률은 0이다.

4.1. Sampling Zero Dynamics Attack

앞에서 살펴본 바에 따르면, 만약 주어진 시스템이 최소위상이어서 zero dynamics가 안정하거나, 혹은 zero dynamics가 없다면 (즉, 상대차수가 $n = \nu$), 해커의 zero dynamics attack은 별 의미가 없게 된다. 그렇다면, 이러한 시스템은 공격으로부터 안전한 것일까?

불행히도 꼭 그렇지만은 않다. 실제 플랜트는 연속시간(continuous-time) 시스템이겠지만 이를 디지털컴퓨터로 제어하기 위해서 우리는 보통 출력을 샘플링하여 이산시간(discrete-time) 신호로 만들며, 또한 구동기단에서는 zero-order hold를 사용하여 이산시간 신호를 연속시간 신호로 만들어 제어를 하곤 한다. 이렇게 되면, 전체 시스템을 이산시간 시스템으로 나타낼 수 있고, 이 경우 이산시간 플랜트는 sampling zero라고 하는 새로운 영점을 갖게 될 수도 있다. 이렇게 새로 만들어진 영점은 플랜트의 상대차수가 2보다 크고 샘플링이 상당히 빠르게 이루어질 경우 언제나 불안정하다는 것을 살펴본 바 있다 [8]. 따라서 이산시간에서 플랜트를 이산시간 normal form으로 나타낸 후 앞서 설명한 것과 동일한 방법으로 이산시간의 zero dynamics attack을 수행할 수 있게 된다. Sampling zero dynamics attack의 실제 예는 [4]에서 찾아볼 수 있다.

4.2. Robust Zero Dynamics Attack

한편, 이상의 논의에서 알게된 사실은, 효과적인 zero dynamics attack을 위해서 해커는 플랜트에 관해 많은 것을 알고 있어야 한다는 것이다. 특히, (8)을 구성하는 행렬 S와 g의 값은 시스템 식별이 충분히 잘 이루어져야 하는데, 통상 시스템 모델과 실제 시스템과는 불확실성에 의한 차이가 존재하므로, 제어기 설계자도 잘 모르는 실제 시스템의 파라메터를 해커가 잘 알 수 있다는 생각은 비현실적일 것이다. 그렇다면, 어차피 해커는 시스템 파라메터를 잘 모를 것이고 따라서 부정확한 시스템 파라메터를 이용한 해커의 공격은 쉽게 발각이 되어, 결국 우리는 크게 걱정할 것이 없게 되는 것일까?

불행히도 꼭 그렇지만은 않다. 시스템 파라메터를 잘 모를때 제어기 설계자가 강인제어기법(robust control)을 활용하여 강인 제어기를 만들 수 있는 것처럼, 해커 역시 강인제어기법을 활용하여 시스템 파라메터를 잘 모르는 상황에서도 효과적으로 은닉 공격을 시행할 수 있다. 하나의 예로, 강인제어기법의 하나인 외란관측기를 해커가 사용할 경우 매우 강력한 공격이 이루어질 수 있음이 알려져 있다 [5].

4.3. Enforced Zero Dynamics Attack

또다른 공격 아이디어 중 하나는, 연속시간 출력 $y(t)$를 일정 시간 간격으로 샘플링 하는 경우, 샘플링을 하는 시점에만 플랜트의 동작이 정상인 것처럼 보이게 하는 공격을 생각해 볼 수 있다. 이는 마치 회전하는 CCTV 밑에서 카메라가 비추지 않는 동안에만 나쁜 일을 하는 범죄수법을 연상시키는데, 동적시스템인 제어시스템에 있어서도 입력의 갯수가 출력보다 많거나 혹은 구동기단의 샘플링 속도가 출력센서의 샘플링 속도보다 빠른 경우에 이러한 범죄수법이 가능하다는 것이 알려져 있다 [6]. 이는 원래 존재하지 않는 영점을 강제로 만드는 것으로 이해할 수도 있어 enforced zero dynamics attack이라 불릴 수 있을 것이다.

5. 어떻게 방어할 것인가?

Zero dynamics attack은 시스템의 고유성질이기 때문에 원리적으로 이들 공격을 효과적으로 발각하는 방법은 없다. 다만, 연속시간 플랜트를 이산시간 제어기로 제어하는 경우에 구동기단에서 사용하는 zero-order hold를 generalized hold로 교체하여, 플랜트의 영점을 안정한 영역으로 옮기는 방법을 사용한다면, zero dynamics attack을 발견할 수는 없지만 설령 그러한 공격이 들어온다 하여도 그 효과를 미미하게 만들 수 있다 [4]. 혹은, 출력신호 y와 제어신호 u, 그리고 제어기 전체를 동형암호(homomorphic encryption)를 사용하여 보호하는 방법 [7]과 같이 근원적으로 공격을 차단하는 아이디어를 활용할 수도 있겠다.

사실 제어시스템을 몰래 공격하는 방법과 이를 효과적으로 방어하는 방법은 아직 많은 연구가 필요한 분야로 최근 제어와 관계된 학계와 산업계를 중심으로 활발한 연구가 이루어지고 있는 분야이다.

참고

이 분야를 소개해 주고 함께 연구해 준 DGIST 고신뢰 CPS 연구센터장 은용순 교수님께 감사하며, 본 편의 LaTeX 조판본은 제어이론연구회 홈페이지에서 구할 수 있다.

REFERENCES

[1] "Cyberphysical Security," *IEEE Control Systems Magazine*, February, 2015.

[2] H. Khalil, *Nonlinear Systems*, 3rd edition, Prentice Hall, 2002.

[3] G. Park, C. Lee, and H. Shim, "On stealthiness of zero-dynamics attacks against uncertain nonlinear systems: A case study with quadruple-tank process," *In Proc. of International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, Hong Kong, 2018.

[4] J. Back, J. Kim, C. Lee, G. Park, and H. Shim, "Enhancement of security against zero dynamics attack via generalized hold," *In Proc. of IEEE Conf. on Decision and Control*, pp. 1350-1355, 2017.

[5] G. Park, C. Lee, H. Shim, Y. Eun, and K. H. Johansson, "Stealthy adversaries against uncertain cyber-physical systems: Threat of robust zero-dynamics attack," to appear at *IEEE Trans. on Automatic Control,* doi:10.1109/TAC.2019.2903429, 2019.

[6] J. Kim, G. Park, H. Shim, and Y. Eun, "A masking attack for asynchronous sampled-data systems via input redundancy," to appear at *IET Control Theory & Applications*, doi:10.1049/iet-cta.2018.6075, 2019

[7] J. Kim, H. Shim, and K. Han, "Comprehensive Introduction to Homomorphic Encryption for Dynamic Feedback Controller via LWE-based Cryptosystem," arXiv:1904.08025, 2019.

[8] 심형보, "알기쉬운 제어이론: Sampling Zero란 무엇인가," 제어로봇시스템학회지, 2019년 3월호

심형보 교수는 2000년 서울대학교 전기공학부에서 박사학위를 받고 미국 산타바바라 소재 캘리포니아 주립대학에서 박사후 과정을 수료 후 현재 서울대학교에서 교수로 재직 중이다. Automatica, IEEE Transactions on Automatic Control 등 저널의 associate editor를 맡은 바 있다.